

14. 4. 2004

日本国特許庁
JAPAN PATENT OFFICE

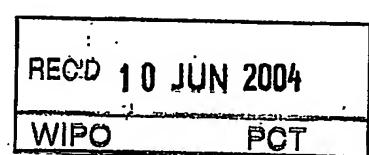
別紙添付の書類に記載されている事項は下記の出願書類に記載されて
いる事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed
with this Office.

出願年月日
Date of Application: 2003年 4月16日

出願番号
Application Number: 特願2003-111978

[ST. 10/C]: [JP2003-111978]



出願人
Applicant(s): 古庄 晋二

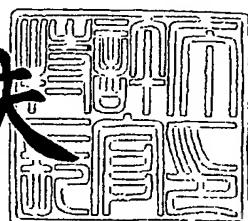


PRIORITY DOCUMENT
SUBMITTED OR TRANSMITTED IN
COMPLIANCE WITH
RULE 17.1(a) OR (b)

2004年 5月27日

特許庁長官
Commissioner,
Japan Patent Office

今井康夫



【書類名】 特許願
【整理番号】 PK030002
【あて先】 特許庁長官 殿
【国際特許分類】 G06F 17/30
【発明者】
【住所又は居所】 神奈川県横浜市神奈川区松見町4丁目1101番地7
コートハウス菊名804号
【氏名】 古庄 晋二
【特許出願人】
【識別番号】 598108515
【氏名又は名称】 古庄 晋二
【代理人】
【識別番号】 100103632
【弁理士】
【氏名又は名称】 齢田 英一郎
【選任した代理人】
【識別番号】 100099715
【弁理士】
【氏名又は名称】 吉田 聰
【手数料の表示】
【予納台帳番号】 058377
【納付金額】 21,000円
【提出物件の目録】
【物件名】 明細書 1
【物件名】 図面 1
【物件名】 要約書 1
【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理システムおよび情報処理方法

【特許請求の範囲】

【請求項1】 それぞれ、メモリ、インターフェース、および、制御装置を有する、複数のメモリモジュールと、

隣接するメモリモジュールのインターフェース間を接続するパケット伝送路であって、一方のメモリモジュールから他方のメモリモジュールにパケットを伝達する第1のパケット伝送路、および、他方のメモリモジュールから一方のメモリモジュールにパケットを伝達する第2のパケット伝送路を備えたパケット伝送路とを備え、

前記メモリモジュールの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されているリスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されるように構成された情報処理システムであって、

各メモリモジュールの制御装置が、前記ポインタ配列のうち、グローバルな情報ブロックの部分集合として、自己の掌握する情報ブロックが、どの位置を占めるかを示すオフセット値を保持するオフセット値記憶手段と、

前記オフセット値に基づき、グローバルな情報ブロックにおけるグローバル順序集合配列を生成するグローバル順序集合配列生成手段と、

隣接するメモリモジュールの間で、一方の伝送路を利用して、自己の、ある項目の値リストをパケット化して送信するパケット送信手段と、

前記パケット送信手段によるパケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された値リストを受信するパケット受信手段と、

受信した値リストを参照して、自己の、当該項目の値リスト中の項目値の相対

的な順位を判定し、当該項目値の順位を、グローバルな情報ブロックにおける順位として、当該項目に関するグローバル値番号配列に収容する順序判定手段とを備え、

各項目について、パケット送信手段によるパケット送信、パケット受理手段によるパケット受理、および、順序判定手段によるグローバル値番号配列への順位の収容を繰り返すように構成されたことを特徴とする情報処理システム。

【請求項2】 前記順序判定手段が、送信したパケットおよび受理したパケットを比較し、重複する値を削除することを特徴とする請求項1に記載の情報処理システム。

【請求項3】 各メモリモジュールの制御装置が、

検索すべき項目に関して、当該項目の値リストと同じサイズのフラグ配列を生成し、検索条件に合致する項目値に対応するフラグ配列中に特定の値を付与するフラグ配列セットアップ手段と、

前記検索すべき項目に関して、順序集合配列が示す位置に対応するポインタ配列中の値を特定し、その後、ポインタ配列中の値が示す位置に対応するフラグ配列中の値を特定することにより、当該順序集合配列中の値に対応するレコードが、検索条件に合致するか否かを判定する検索条件判定手段と、

検索条件に合致する順序集合の値、および、対応するグローバル順序集合の値を、それぞれ、第2の順序集合配列および第2のグローバル順序集合配列に収容するローカル検索手段とを備え、

前記パケット送信手段が、一方の伝送路を利用して、前記第2のグローバル順序集合配列をパケット化して送信し、かつ、前記パケット受理手段が、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル順序集合配列を受理し、さらに、

受理した第2のグローバル順序集合配列を参照して、自己の、グローバル順序集合配列中の値の相対的な順位を判定して、当該順位を、第3のグローバル順序集合配列に収容する第2の順序判定手段を備え、

パケット送信手段によるパケット送信、パケット受理手段によるパケット受理、および、第2の順序判定手段による第3のグローバル順序集合配列への順位の

収容を繰り返すことにより、前記第3のグローバル順序集合配列の値によって、検索条件に合致するレコードの順位が規定されることを特徴とする請求項1または2に記載の情報処理システム。

【請求項4】 各メモリモジュールの制御装置が、

集計すべき項目に関して、当該項目の値リストのサイズを乗じたサイズの論理座標配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記論理座標配列の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得するカウントアップ手段を備え、

前記パケット送信手段が、前記カウントアップ手段によるカウントアップがなされた論理座標配列を、一方の伝送路を利用して、パケット化して送信し、各メモリモジュールにおいて、同一の論理座標配列のカウントアップおよび一方の伝送路を利用した送信を、順次実行することにより、前記論理座標配列に、グローバルな各項目の項目値の組ごとのレコード数が収容され、かつ、

各メモリモジュールにおいて、前記パケット受信手段およびパケット送信手段が、カウントアップが終了した論理座標配列の受理および記憶を、並びに、一方の伝送路を利用した送信を順次実行することを特徴とする請求項1ないし3の何れか一項に記載の情報処理システム。

【請求項5】 前記カウントアップ手段が、集計すべき項目に関して、当該項目の値リストのサイズを乗じた多次元のカウントアップ配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記カウントアップ配列中の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得し、かつ、カウントアップ配列中の位置とのマッピングがされた論理座標配列中、当該カウントアップ配列中の値を、前記マッピングにしたがって配置するように構成されたことを特徴とする請求項4に記載の情報処理システム。

【請求項6】 各メモリモジュールの制御装置が、

ソートすべき項目に関して、当該項目の値リストと同じサイズの存在数配列を生成し、値リスト中の項目値のそれぞれを指定する、前記順序集合配列の値の数

を配置する存在数配列生成手段と、

前記存在数配列中の値を累計して、メモリモジュール内でソートされた際の、
対応する項目値をもつレコードの先頭位置を示す累計数を算出し、当該累計数を
累計数配列中に配置する累計数配列生成手段と、

第2のグローバル値番号配列、第4のグローバル順序集合配列および第3の順
序集合配列を生成し、順序集合配列の値が示す項目値に対応する累計数配列中の
累計数に基づき、前記第2のグローバル値番号配列中、前記累計数が示す位置に
、前記項目値に対応するグローバル値番号を配置し、かつ、前記第3の順序集合
配列、および、前記第4のグローバル順序集合配列中、前記累計数が示す位置に
、前記順序集合配列の値、および、対応するグローバル順序集合配列の値を、そ
れぞれ配置する、ローカルソート手段とを備え、

前記パケット送信手段が、一方の伝送路を利用して、少なくとも、第2のグロ
ーバル値番号配列をパケット化して送信し、かつ、前記パケット受理手段が、並
列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された
第2のグローバル値番号配列を受理し、さらに、

受理した第2のグローバル値番号配列を参照して、自己の、第2のグローバル
値番号配列中の値の相対的な順位を判定して、当該順位を、第5のグローバル順
序集合配列に収容する第3の順序判定手段を備え、

パケット送信手段によるパケット送信、パケット受理手段によるパケット受理
、および、第3の順序判定手段による第5のグローバル順序集合配列への順位の
収容を繰り返すことにより、前記第5のグローバル順序集合配列の値によって、
ソートされたレコードの順位が規定されることを特徴とする請求項1ないし5の
何れか一項に記載の情報処理システム。

【請求項7】 前記パケット送信手段が、第2のグローバル値番号配列の値お
よび第4のグローバル順序集合配列の値を組とすることで、前記第2のグローバ
ル値番号配列および第4のグローバル順序集合配列をパケット化して送信し、か
つ、前記パケット受理手段は、他のメモリモジュールの、パケット化された、前
記第2のグローバル値番号配列および第4のグローバル順序集合配列を受理し、

前記第3の順序判定手段が、自己の、第2のグローバル値番号配列の値と、他

のメモリモジュールの第2のグローバル番号配列の値とが同一であるときに、それぞれの値の組をなす、第4のグローバル順序集合配列の値を比較することで、順位を判定することを特徴とする請求項6に記載の情報処理システム。

【請求項8】 前記メモリモジュールの制御装置が、前記配列として利用するためのレジスタ群を有し、前記配列を利用した演算は、メモリをアクセスすることなく実行されることを特徴とする請求項1ないし7の何れか一項に記載の情報処理システム。

【請求項9】 それぞれ、メモリ、インターフェース、および、制御装置を有する、複数のメモリモジュールと、

隣接するメモリモジュールのインターフェース間を接続するパケット伝送路であって、一方のメモリモジュールから他方のメモリモジュールにパケットを伝達する第1のパケット伝送路、および、他方のメモリモジュールから一方のメモリモジュールにパケットを伝達する第2のパケット伝送路を備えたパケット伝送路とを備え、

前記メモリモジュールの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されるように構成された情報処理システムにおいて、

各メモリモジュールにおいて、前記ポインタ配列のうち、グローバルな情報ブロックの部分集合として、自己の掌握する情報ブロックが、どの位置を占めるかを示すオフセット値を保持するオフセット値記憶ステップと、

前記オフセット値に基づき、グローバルな情報ブロックにおけるグローバル順序集合配列を生成するグローバル順序集合配列生成ステップと、

隣接するメモリモジュールの間で、一方の伝送路を利用して、自己の、ある項目の値リストをパケット化して送信するパケット送信ステップと、

前記パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された値リストを受理するパケット受理ステップと、

受理した値リストを参照して、自己の、当該項目の値リスト中の項目値の相対的な順位を判定し、当該項目値の順位を、グローバルな情報ロックにおける順位として、当該項目に関するグローバル値番号配列に収容する順序判定ステップとを備え、

各項目について、パケット送信ステップ、パケット受理ステップ、および、順序判定ステップにおけるグローバル値番号配列への順位の収容を繰り返すように構成されたことを特徴とする情報処理方法。

【請求項10】 前記順序判定ステップにおいて、送信したパケットおよび受理したパケットを比較し、重複する値を削除することを特徴とする請求項9に記載の情報処理方法。

【請求項11】 各メモリモジュールにおいて、

検索すべき項目に関して、当該項目の値リストと同じサイズのフラグ配列を生成し、検索条件に合致する項目値に対応するフラグ配列中に特定の値を付与するフラグ配列セットアップステップと、

前記検索すべき項目に関して、順序集合配列が示す位置に対応するポインタ配列中の値を特定し、その後、ポインタ配列中の値が示す位置に対応するフラグ配列中の値を特定することにより、当該順序集合配列中の値に対応するレコードが、検索条件に合致するか否かを判定する検索条件判定ステップと、

検索条件に合致する順序集合の値、および、対応するグローバル順序集合の値を、それぞれ、第2の順序集合配列および第2のグローバル順序集合配列に収容するローカル検索ステップと、

一方の伝送路を利用して、前記第2のグローバル順序集合配列をパケット化して送信する第2のパケット送信ステップと、

前記パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル順序集合配列を受理する第2のパケット受理ステップと、

受理した第2のグローバル順序集合配列を参照して、自己の、グローバル順序

集合配列中の値の相対的な順位を判定して、当該順位を、第3のグローバル順序集合配列に収容する第2の順序判定ステップとを備え、

第2のパケット送信ステップ、第2のパケット受理ステップ、および、第2の順序判定ステップにおける第3のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第3のグローバル順序集合配列の値によって、検索条件に合致するレコードの順位が規定されることを特徴とする請求項9または10に記載の情報処理方法。

【請求項12】 各メモリモジュールにおいて、

集計すべき項目に関して、当該項目の値リストのサイズを乗じたサイズの論理座標配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記論理座標配列の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得するカウントアップステップと、

カウントアップがなされた論理座標配列を、一方の伝送路を利用して、パケット化して送信する第3のパケット送信ステップとを備え、

各メモリモジュールにおいて、同一の論理座標配列に対するカウントアップステップ、および、一方の伝送路を利用した送信ステップを、順次実行することにより、前記論理座標配列に、グローバルな各項目の項目値の組ごとのレコード数が収容され、さらに、

各メモリモジュールにおいて、カウントアップが終了した論理座標配列を受理および記憶する第3のパケット受理ステップと、

受理した論理座標配列を、一方の伝送路を利用して送信する第4のパケット送信ステップとを備えたことを特徴とする請求項9ないし11の何れか一項に記載の情報処理方法。

【請求項13】 前記カウントアップステップにおいて、

集計すべき項目に関して、当該項目の値リストのサイズを乗じた多次元のカウントアップ配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記カウントアップ配列中の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得し、かつ、

カウントアップ配列中の位置とのマッピングがされた論理座標配列中、当該カウントアップ配列中の値を、前記マッピングにしたがって配置するように構成されたことを特徴とする請求項12に記載の情報処理方法。

【請求項14】 各メモリモジュールにおいて、

ソートすべき項目に関して、当該項目の値リストと同じサイズの存在数配列を生成し、値リスト中の項目値のそれぞれを指定する、前記順序集合配列の値の数を配置する存在数配列生成ステップと、

前記存在数配列中の値を累計して、メモリモジュール内でソートされた際の、対応する項目値をもつレコードの先頭位置を示す累計数を算出し、当該累計数を累計数配列中に配置する累計数配列生成ステップと、

第2のグローバル値番号配列、第4のグローバル順序集合配列および第3の順序集合配列を生成し、順序集合配列の値が示す項目値に対応する累計数配列中の累計数に基づき、前記第2のグローバル値番号配列中、前記累計数が示す位置に、前記項目値に対応するグローバル値番号を配置し、かつ、前記第3の順序集合配列、および、前記第4のグローバル順序集合配列中、前記累計数が示す位置に、前記順序集合配列の値、および、対応するグローバル順序集合配列の値を、それぞれ配置する、ローカルソートステップと、

一方の伝送路を利用して、少なくとも、第2のグローバル値番号配列をパケット化して送信する第5のパケット送信ステップと、

パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル値番号配列を受理する第4のパケット受理ステップと、

受理した第2のグローバル値番号配列を参照して、自己の、第2のグローバル値番号配列中の値の相対的な順位を判定して、当該順位を、第5のグローバル順序集合配列に収容する第3の順序判定ステップとを備え、

パケット送信ステップ、パケット受理ステップ、および、第3の順序判定ステップにおける第5のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第5のグローバル順序集合配列の値によって、ソートされたレコードの順位が規定されることを特徴とする請求項9ないし14の何れか一項に記載の情

報処理方法。

【請求項15】 前記第5のパケット送信ステップにおいて、第2のグローバル値番号配列の値および第4のグローバル順序集合配列の値を組とすることで、前記第2のグローバル値番号配列および第4のグローバル順序集合配列をパケット化して送信し、

前記第4のパケット受理ステップにおいて、他のメモリモジュールの、パケット化された、前記第2のグローバル値番号配列および第4のグローバル順序集合配列を受理し、

前記第3の順序判定ステップにおいて、自己の、第2のグローバル値番号配列の値と、他のメモリモジュールの第2のグローバル番号配列の値とが同一であるときに、それぞれの値の組をなす、第4のグローバル順序集合配列の値を比較することで、順位を判定することを特徴とする請求項14に記載の情報処理方法。

【請求項16】 それぞれ、メモリおよび制御装置を有する複数の情報処理ユニットを備え、

前記情報処理ユニットの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを保持し、

各メモリモジュールが保持する表形式データの集合体により、グローバルな表形式データが形成されるような情報処理システムであって、

前記各情報処理ユニットが、

前記グローバルな表形式データにおける各レコードの順位を示す値を収容するグローバル順序集合配列と、

制御装置により受理された順位を指定する命令にしたがって、前記グローバル順序集合配列中の値を特定し、その値が示すレコードを取り出すレコード取り出し手段とを備えたことを特徴とする情報処理システム。

【請求項17】 前記情報処理ユニットが、

当該情報処理ユニット内でのソート順を反映するため、レコードを特定する値が入れ替えられた他の順序集合配列を有し、

前記グローバル順序集合配列において、他の順序集合配列中の値が示すレコード

ドの、前記グローバルな表形式データにおけるソート順を示すように、その順位を示す値が再配置されたことを特徴とする請求項16に記載の情報処理システム。

【請求項18】 前記情報処理ユニットが、

前記グローバル順序集合配列において、前記情報処理ユニット内でソートされたレコードの、前記グローバルな表形式データにおけるソート順を示すように、その順位を示す値が再配置されたことを特徴とする請求項16に記載の情報処理システム。

【請求項19】 前記情報処理ユニットの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、

各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されたことを特徴とする請求項16に記載の情報処理システム。

【請求項20】 それぞれ、メモリおよび制御装置を有する複数の情報処理ユニットを備え、

前記情報処理ユニットの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されるような情報処理システムであつて、

前記情報処理ユニットが、

グローバルな情報ブロックにおける項目値の順位を示す値を収容するグローバル値番号配列と、

制御装置により受理された順位を指定する命令にしたがって、前記グローバル値番号配列中の値を特定し、その値が示す、値リスト中の項目値を取り出す項目値取り出し手段とを備えたことを特徴とする情報処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、 SIMD (Single Instruction Stream, Multiple Data Stream) を実現可能な並列コンピュータのアーキテクチャを採用した情報処理システムに関する。

【0002】

【従来の技術】

社会全体のさまざまな場所にコンピュータが導入され、インターネットをはじめとするネットワークが浸透した今日では、そこそこで、大規模なデータが蓄積されるようになった。このような大規模データを処理するには、膨大な計算が必要で、そのために並列処理を導入しようと試みるのは自然である。

【0003】

並列処理アーキテクチャは「共有メモリ型」と「分散メモリ型」に大別される。前者（「共有メモリ型」）は、複数のプロセッサが1つの巨大なメモリ空間を共有する方式である。この方式では、プロセッサ群と共有メモリ間のトラフィックがボトルネックとなるので、百を越えるプロセッサを用いて現実的なシステムを構築することは容易ではない。したがって、例えば10億個の浮動小数点変数の平方根を計算する際、単一CPUに対する加速比はせいぜい100倍ということになる。経験的には、30倍程度が上限である。

【0004】

後者（「分散メモリ型」）は、各プロセッサがそれぞれローカルなメモリを持ち、これらを結合してシステムを構築する。この方式では、数百～数万ものプロセッサを組み込んだハードウェアシステムの設計が可能である。したがって、上記10億個の浮動小数点変数の平方根を計算する際の単一CPUに対する加速比を、数百～数万倍とすることが可能である。しかしながら、後者においても、後

述するいくつかの課題が存在する。

【0005】

【特許文献1】 国際公開第WO00/10103号パンフレット（第3図および第4図）

【0006】

【発明が解決しようとする課題】

[第1の課題：巨大配列の分掌管理]

「分散メモリ型」の第1の課題は、データの分掌管理の問題である。

巨大なデータ（一般的には配列なので、以降、配列で説明する）は、1つのプロセッサの所有するローカルメモリに収容できるものではなく、必然的に複数のローカルメモリに分掌管理される。効率的かつ柔軟な分掌管理メカニズムを導入しないと、プログラムの開発および実行に際してさまざまな障害を抱え込むことになることは明らかである。

【0007】

[第2の課題：プロセッサ間通信の効率の低さ]

分散メモリ型システムの各プロセッサが、巨大配列にアクセスしようとすると、自己の所有するローカルメモリ上の配列要素に対しては速やかにアクセスできるものの、他のプロセッサが所有する配列要素へのアクセスはプロセッサ間通信を必須とする。このプロセッサ間通信はローカルメモリとの通信に比べ、極端にパフォーマンスが低く、最低でも100クロックかかると言われている。このため、ソート実施時には、巨大配列全域にわたる参照が実施され、プロセッサ間通信が多発するため、パフォーマンスが極端に低下する。

【0008】

この問題点につき、より具体的に説明を加える。1999年現在、パソコンは、1～数個のCPUを用いて、「共有メモリ型」として構成されている。このパソコンに使用される標準的なCPUは、メモリバスの5～6倍程度の内部クロックで動作し、その内部に自動的な並列実行機能やパイプライン処理機能が装備されており、およそ1データを1クロック（メモリバス）で処理できる。

【0009】

このため、「分散メモリ型」のマルチプロセッサシステムでは、プロセッサ数は多いのに、シングルプロセッサ（共有メモリ型）よりも100倍遅くなることになりかねない。

【0010】

[第3の課題：プログラムの供給]

「分散メモリ型」の第3の課題は、多数のプロセッサにどうやってプログラムを供給するか、という問題である。

非常に多数のプロセッサに、別々のプログラムをロードし、全体を協調動作させる方式（MIMD：Multiple Instruction Stream, Multiple Data Stream）では、プログラムの作成、コンパイル、配信のために多大な負荷を要する。

その一方、多数のプロセッサを同一のプログラムで動作させる方式（SIMD：Single Instruction Stream, Multiple Data Stream）では、プログラムの自由度が減少し、所望の結果をもたらすプログラムが開発できない事態も想定される。

【0011】

本発明は、「分散メモリ型」の上記第1ないし3の課題を解決する方法およびコンピュータアーキテクチャを提供する。

【0012】

ところで、本発明者は、表形式データを記憶するために、項目ごとの情報ブロックを形成し、当該情報ブロックに、項目値を記憶した値リスト、および、当該値リストを指定するための値（ポインタ値）を、レコードごとに記憶したポインタ配列を設け、レコード番号から、ポインタ配列および値リストを順次特定していくことにより、表形式のビューを取得できるような構造および処理方法を考案している（特許文献1参照）。この構造において、レコード数が増大するのにしたがって、上記値リストやポインタ配列、特に、ポインタ配列は非常に大きくなるため、これを、複数のメモリで分掌した上で、单一命令により、検索、集計、ソートなどの処理が実行できるのが望ましい。

【0013】

そこで、本発明は、分散メモリ型において、单一命令により種々のメモリに記

憶された配列中の要素を入出力し、処理と通信を統合することで著しく高速な並列処理を実現可能なコンピューターアーキテクチャを提供することを目的とする。

【0014】

【課題を解決するための手段】

本発明の目的は、それぞれ、メモリ、インターフェース、および、制御装置を有する、複数のメモリモジュールと、

隣接するメモリモジュールのインターフェース間を接続するパケット伝送路であつて、一方のメモリモジュールから他方のメモリモジュールにパケットを伝達する第1のパケット伝送路、および、他方のメモリモジュールから一方のメモリモジュールにパケットを伝達する第2のパケット伝送路を備えたパケット伝送路とを備え、

前記メモリモジュールの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されるように構成された情報処理システムであつて、

各メモリモジュールの制御装置が、前記ポインタ配列のうち、グローバルな情報ブロックの部分集合として、自己の掌握する情報ブロックが、どの位置を占めるかを示すオフセット値を保持するオフセット値記憶手段と、

前記オフセット値に基づき、グローバルな情報ブロックにおけるグローバル順序集合配列を生成するグローバル順序集合配列生成手段と、

隣接するメモリモジュールの間で、一方の伝送路を利用して、自己の、ある項目の値リストをパケット化して送信するパケット送信手段と、

前記パケット送信手段によるパケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された値リストを受理するパケット受理手段と、

受理した値リストを参照して、自己の、当該項目の値リスト中の項目値の相対的な順位を判定し、当該項目値の順位を、グローバルな情報ブロックにおける順位として、当該項目に関するグローバル値番号配列に収容する順序判定手段とを備え、

各項目について、パケット送信手段によるパケット送信、パケット受信手段によるパケット受信、および、順序判定手段によるグローバル値番号配列への順位の収容を繰り返すように構成されたことを特徴とする情報処理システムにより達成される。

【0015】

本発明によれば、第1のパケット伝送路および第2のパケット伝送路に、並列的に、パケットを送出し、それぞれのPMMにおいて、他のPMMの値リストの順位を考慮した上で、自分自身の掌握するローカルな値リストの項目値の順位を決定することができる。したがって、それぞれのPMMにおいて、グローバルな表形式データとして、自己が掌握する部分集合の位置ないし順位を適切に把握することが可能となる。このように位置ないし順位を把握しておくことで、後述する検索、クロス集計およびソートの処理が円滑に実現することができる。

【0016】

好ましい実施態様においては、前記順序判定手段が、送信したパケットおよび受信したパケットを比較し、重複する値を削除する。

【0017】

別の好ましい実施態様においては、各メモリモジュールの制御装置が、検索すべき項目に関して、当該項目の値リストと同じサイズのフラグ配列を生成し、検索条件に合致する項目値に対応するフラグ配列中に特定の値を付与するフラグ配列セットアップ手段と、

前記検索すべき項目に関して、順序集合配列が示す位置に対応するポインタ配列中の値を特定し、その後、ポインタ配列中の値が示す位置に対応するフラグ配列中の値を特定することにより、当該順序集合配列中の値に対応するレコードが、検索条件に合致するか否かを判定する検索条件判定手段と、

検索条件に合致する順序集合の値、および、対応するグローバル順序集合の値

を、それぞれ、第2の順序集合配列および第2のグローバル順序集合配列に収容するローカル検索手段とを備え、

前記パケット送信手段が、一方の伝送路を利用して、前記第2のグローバル順序集合配列をパケット化して送信し、かつ、前記パケット受理手段が、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル順序集合配列を受理し、さらに、

受理した第2のグローバル順序集合配列を参照して、自己の、グローバル順序集合配列中の値の相対的な順位を判定して、当該順位を、第3のグローバル順序集合配列に収容する第2の順序判定手段を備え、

パケット送信手段によるパケット送信、パケット受理手段によるパケット受理、および、第2の順序判定手段による第3のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第3のグローバル順序集合配列の値によって、検索条件に合致するレコードの順位が規定される。

【0018】

別の好ましい実施態様においては、各メモリモジュールの制御装置が、集計すべき項目に関して、当該項目の値リストのサイズを乗じたサイズの論理座標配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポイント配列中の値の組に対応する、前記論理座標配列の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得するカウントアップ手段を備え、

前記パケット送信手段が、前記カウントアップ手段によるカウントアップがなされた論理座標配列を、一方の伝送路を利用して、パケット化して送信し、各メモリモジュールにおいて、同一の論理座標配列のカウントアップおよび一方の伝送路を利用した送信を、順次実行することにより、前記論理座標配列に、グローバルな各項目の項目値の組ごとのレコード数が収容され、かつ、

各メモリモジュールにおいて、前記パケット受信手段およびパケット送信手段が、カウントアップが終了した論理座標配列の受理および記憶を、並びに、一方の伝送路を利用した送信を順次実行する。

【0019】

より好ましくは、カウントアップ手段は、集計すべき項目に関して、当該項目の値リストのサイズを乗じた多次元のカウントアップ配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記カウントアップ配列中の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得し、かつ、カウントアップ配列中の位置とのマッピングがされた論理座標配列中、当該カウントアップ配列中の値を、前記マッピングにしたがって配置するように構成されている。

【0020】

別の好ましい実施態様においては、各メモリモジュールの制御装置が、ソートすべき項目に関して、当該項目の値リストと同じサイズの存在数配列を生成し、値リスト中の項目値のそれぞれを指定する、前記順序集合配列の値の数を配置する存在数配列生成手段と、

前記存在数配列中の値を累計して、メモリモジュール内でソートされた際の、対応する項目値をもつレコードの先頭位置を示す累計数を算出し、当該累計数を累計数配列中に配置する累計数配列生成手段と、

第2のグローバル値番号配列、第4のグローバル順序集合配列および第3の順序集合配列を生成し、順序集合配列の値が示す項目値に対応する累計数配列中の累計数に基づき、前記第2のグローバル値番号配列中、前記累計数が示す位置に、前記項目値に対応するグローバル値番号を配置し、かつ、前記第3の順序集合配列、および、前記第4のグローバル順序集合配列中、前記累計数が示す位置に、前記順序集合配列の値、および、対応するグローバル順序集合配列の値を、それぞれ配置する、ローカルソート手段とを備え、

前記パケット送信手段が、一方の伝送路を利用して、少なくとも、第2のグローバル値番号配列をパケット化して送信し、かつ、前記パケット受理手段が、並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル値番号配列を受理し、さらに、

受理した第2のグローバル値番号配列を参照して、自己の、第2のグローバル値番号配列中の値の相対的な順位を判定して、当該順位を、第5のグローバル順序集合配列に収容する第3の順序判定手段を備え、

パケット送信手段によるパケット送信、パケット受理手段によるパケット受理、および、第3の順序判定手段による第5のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第5のグローバル順序集合配列の値によって、ソートされたレコードの順位が規定される。

【0021】

より好ましくは、前記パケット送信手段が、第2のグローバル値番号配列の値および第4のグローバル順序集合配列の値を組とすることで、前記第2のグローバル値番号配列および第4のグローバル順序集合配列をパケット化して送信し、かつ、前記パケット受理手段は、他のメモリモジュールの、パケット化された、前記第2のグローバル値番号配列および第4のグローバル順序集合配列を受理し、

前記第3の順序判定手段が、自己の、第2のグローバル値番号配列の値と、他のメモリモジュールの第2のグローバル番号配列の値とが同一であるときに、それぞれの値の組をなす、第4のグローバル順序集合配列の値を比較することで、順位を判定するように構成される。

【0022】

さらに別の好ましい実施態様においては、前記メモリモジュールの制御装置が、前記配列として利用するためのレジスタ群を有し、前記配列を利用した演算は、メモリをアクセスすることなく実行される。配列のほか、オフセット値などもレジスタに記憶することができる。

【0023】

上述した構成における各手段、たとえば、順序集合配列生成手段、順序判定手段、フラグ配列セットアップ手段、検査条件判定手段、ローカル検索手段などは、メモリモジュール中の制御装置により実現される。

【0024】

また、本発明の目的は、それぞれ、メモリ、インターフェース、および、制御装置を有する、複数のメモリモジュールと、

隣接するメモリモジュールのインターフェース間を接続するパケット伝送路であって、一方のメモリモジュールから他方のメモリモジュールにパケットを伝達す

る第1のパケット伝送路、および、他方のメモリモジュールから一方のメモリモジュールにパケットを伝達する第2のパケット伝送路を備えたパケット伝送路とを備え、

前記メモリモジュールの各々のメモリが、

各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ロックを保持し、各メモリにて保持された情報ロックの集合体により、グローバルな情報ロックが形成されるように構成された情報処理システムにおいて、

各メモリモジュールにおいて、前記ポインタ配列のうち、グローバルな情報ロックの部分集合として、自己の掌握する情報ロックが、どの位置を占めるかを示すオフセット値を保持するオフセット値記憶ステップと、

前記オフセット値に基づき、グローバルな情報ロックにおけるグローバル順序集合配列を生成するグローバル順序集合配列生成ステップと、

隣接するメモリモジュールの間で、一方の伝送路を利用して、自己の、ある項目の値リストをパケット化して送信するパケット送信ステップと、

前記パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された値リストを受理するパケット受理ステップと、

受理した値リストを参照して、自己の、当該項目の値リスト中の項目値の相対的な順位を判定し、当該項目値の順位を、グローバルな情報ロックにおける順位として、当該項目に関するグローバル値番号配列に収容する順序判定ステップとを備え、

各項目について、パケット送信ステップ、パケット受理ステップ、および、順序判定ステップにおけるグローバル値番号配列への順位の収容を繰り返すように構成されたことを特徴とする情報処理方法により達成される。

【0025】

好ましい実施態様においては、前記順序判定ステップにおいて、送信したパケ

ットおよび受理したパケットを比較し、重複する値を削除する。

別の好ましい実施態様においては、各メモリモジュールにおいて、

検索すべき項目に関して、当該項目の値リストと同じサイズのフラグ配列を生成し、検索条件に合致する項目値に対応するフラグ配列中に特定の値を付与するフラグ配列セットアップステップと、

前記検索すべき項目に関して、順序集合配列が示す位置に対応するポインタ配列中の値を特定し、その後、ポインタ配列中の値が示す位置に対応するフラグ配列中の値を特定することにより、当該順序集合配列中の値に対応するレコードが、検索条件に合致するか否かを判定する検索条件判定ステップと、

検索条件に合致する順序集合の値、および、対応するグローバル順序集合の値を、それぞれ、第2の順序集合配列および第2のグローバル順序集合配列に収容するローカル検索ステップと、

一方の伝送路を利用して、前記第2のグローバル順序集合配列をパケット化して送信する第2のパケット送信ステップと、

前記パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル順序集合配列を受理する第2のパケット受理ステップと、

受理した第2のグローバル順序集合配列を参照して、自己の、グローバル順序集合配列中の値の相対的な順位を判定して、当該順位を、第3のグローバル順序集合配列に収容する第2の順序判定ステップとを備え、

第2のパケット送信ステップ、第2のパケット受理ステップ、および、第2の順序判定ステップにおける第3のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第3のグローバル順序集合配列の値によって、検索条件に合致するレコードの順位が規定される。

【0026】

別の好ましい実施態様において、各メモリモジュールにおいて、

集計すべき項目に関して、当該項目の値リストのサイズを乗じたサイズの論理座標配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記論理座標配列の値をカウントアップすることに

より、各項目の項目値の組ごとの、レコード数を取得するカウントアップステップと、

カウントアップがなされた論理座標配列を、一方の伝送路を利用して、パケット化して送信する第3のパケット送信ステップとを備え、

各メモリモジュールにおいて、同一の論理座標配列に対するカウントアップステップ、および、一方の伝送路を利用した送信ステップを、順次実行することにより、前記論理座標配列に、グローバルな各項目の項目値の組ごとのレコード数が収容され、さらに、

各メモリモジュールにおいて、カウントアップが終了した論理座標配列を受理および記憶する第3のパケット受理ステップと、

受理した論理座標配列を、一方の伝送路を利用して送信する第4のパケット送信ステップとを備えている。

【0027】

より好ましくは、前記カウントアップステップにおいて、

集計すべき項目に関して、当該項目の値リストのサイズを乗じた多次元のカウントアップ配列を生成し、順序集合配列中の値が示す、前記集計すべき項目のポインタ配列中の値の組に対応する、前記カウントアップ配列中の値をカウントアップすることにより、各項目の項目値の組ごとの、レコード数を取得し、かつ、カウントアップ配列中の位置とのマッピングがされた論理座標配列中、当該カウントアップ配列中の値を、前記マッピングにしたがって配置するように構成されている。

【0028】

また、別の好ましい実施態様においては、各メモリモジュールにおいて、

ソートすべき項目に関して、当該項目の値リストと同じサイズの存在数配列を生成し、値リスト中の項目値のそれぞれを指定する、前記順序集合配列の値の数を配置する存在数配列生成ステップと、

前記存在数配列中の値を累計して、メモリモジュール内でソートされた際の、対応する項目値をもつレコードの先頭位置を示す累計数を算出し、当該累計数を累計数配列中に配置する累計数配列生成ステップと、

第2のグローバル値番号配列、第4のグローバル順序集合配列および第3の順序集合配列を生成し、順序集合配列の値が示す項目値に対応する累計数配列中の累計数に基づき、前記第2のグローバル値番号配列中、前記累計数が示す位置に、前記項目値に対応するグローバル値番号を配置し、かつ、前記第3の順序集合配列、および、前記第4のグローバル順序集合配列中、前記累計数が示す位置に、前記順序集合配列の値、および、対応するグローバル順序集合配列の値を、それぞれ配置する、ローカルソートステップと、

一方の伝送路を利用して、少なくとも、第2のグローバル値番号配列をパケット化して送信する第5のパケット送信ステップと、

パケット送信と並列的に、他方の伝送路を利用して、他のメモリモジュールの、パケット化された第2のグローバル値番号配列を受理する第4のパケット受理ステップと、

受理した第2のグローバル値番号配列を参照して、自己の、第2のグローバル値番号配列中の値の相対的な順位を判定して、当該順位を、第5のグローバル順序集合配列に収容する第3の順序判定ステップとを備え、

パケット送信ステップ、パケット受理ステップ、および、第3の順序判定ステップにおける第5のグローバル順序集合配列への順位の収容を繰り返すことにより、前記第5のグローバル順序集合配列の値によって、ソートされたレコードの順位が規定される。

【0029】

より好ましくは、前記第5のパケット送信ステップにおいて、第2のグローバル値番号配列の値および第4のグローバル順序集合配列の値を組とすることで、前記第2のグローバル値番号配列および第4のグローバル順序集合配列をパケット化して送信し、

前記第4のパケット受理ステップにおいて、他のメモリモジュールの、パケット化された、前記第2のグローバル値番号配列および第4のグローバル順序集合配列を受理し、

前記第3の順序判定ステップにおいて、自己の、第2のグローバル値番号配列の値と、他のメモリモジュールの第2のグローバル番号配列の値とが同一である

ときに、それぞれの値の組をなす、第4のグローバル順序集合配列の値を比較することで、順序を判定する。

【0030】

さらに、本発明の目的は、それぞれ、メモリおよび制御装置を有する複数の情報処理ユニットを備え、前記情報処理ユニットの各々のメモリが、各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを保持し、各メモリモジュールが保持する表形式データの集合体により、グローバルな表形式データが形成されるような情報処理システムであって、

前記各情報処理ユニットが、前記グローバルな表形式データにおける各レコードの順位を示す値を収容するグローバル順序集合配列と、制御装置により受理された順位を指定する命令にしたがって、前記グローバル順序集合配列中の値を特定し、その値が示すレコードを取り出すレコード取り出し手段とを備えたことを特徴とする情報処理システムによっても達成される。

【0031】

本発明によれば、PMM、パーソナルコンピュータ、サーバなどを含む情報処理ユニットにローカルな表形式データを分掌把握させ、ローカルな検索や集計を情報処理ユニット単独で実行させることもでき、かつ、グローバル順序集合配列を備えることにより、グローバルな表形式データの検索などを実現することも可能となる。なお、単一のパーソナルコンピュータやサーバが単一の情報処理ユニットに対応しても良いし、単一のパーソナルコンピュータやサーバに、複数の情報処理ユニットが含まれるような構成を採用しても良い。

【0032】

好ましい実施態様においては、前記情報処理ユニットが、当該情報処理ユニット内でのソート順を反映するため、レコードを特定する値が入れ替えられた他の順序集合配列を有し、前記グローバル順序集合配列において、他の順序集合配列中の値が示すレコードの、前記グローバルな表形式データにおけるソート順を示すように、その順位を示す値が再配置される。このグローバル順序集合配列に再配置された値は昇順となる。

【0033】

或いは、前記情報処理ユニットが、前記グローバル順序集合配列において、前記情報処理ユニット内でソートされたレコードの、前記グローバルな表形式データにおけるソート順を示すように、その順位を示す値が再配置されるような構成であっても良い。ここでも、グローバル順序集合配列に再配置された値は昇順となる。このように、本発明は、レコードを特定する値をソートして、これを他の順序集合配列に収容したような形態にも適用でき、また、レコード自体をソートにより並べ替えるような形態にも適用できる。

【0034】

別の好ましい実施態様においては、前記情報処理ユニットの各々のメモリが、各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成される。

【0035】

さらに、本発明の目的は、それぞれ、メモリおよび制御装置を有する複数の情報処理ユニットを備え、前記情報処理ユニットの各々のメモリが、各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成されるような情報処理システムであって、

【0036】

前記情報処理ユニットが、グローバルな情報ブロックにおける項目値の順位を示す値を収容するグローバル値番号配列と、制御装置により受理された順位を指定する命令にしたがって、前記グローバル値番号配列中の値を特定し、その値が示す、値リスト中の項目値を取り出す項目値取り出し手段とを備えたことを特徴

とする情報処理システムによっても達成される。

【0037】

【発明の実施の形態】

【ハードウェア構成】

以下、添付図面を参照して、本発明の実施の形態につき説明を加える。図1は、本発明の実施の形態にかかる情報処理システムの概略を示すブロックダイヤグラムである。図1に示すように、この実施の形態においては、複数のプロセッサ付きメモリモジュール（以下、「PMM」と称する）12-0、12-1、12-2、・・・がリング状に配置され、隣接するメモリモジュール間を、時計回りにデータを伝達する第1のバス（たとえば、符号14-0、14-1参照）、および、反時計回りにデータを伝達する第2のバス（たとえば、符号16-0、16-1参照）が接続している。第1のバスおよび第2のバスでは、PMM間のパケット通信が実行される。本実施の形態において、このパケット通信が実行される伝送路（パケット伝送路）を、第1のバスおよび第2のバスと称する。

【0038】

図2は、PMM12の構造の一例を示す図である。図2に示すように、PMM12は、命令にしたがったメモリのアクセス、演算の実行などを制御する制御回路20と、バスインターフェース（I/F）22と、メモリ24とを備えている。

メモリ24は、複数のバンクBANK0、1、・・・、n（符号26-0、・・・、n）を有し、それぞれに、後述する所定の配列を記憶できるようになっている。

また、制御回路20は、外部の他のコンピュータ等とのデータ授受が可能である。また、他のコンピュータが、バスアービトレーションにより、メモリの所望のバンクにアクセスできるようにしても良い。

【0039】

【データの記憶構造】

図3は、表形式データの一例を示す図である。このように、表形式のデータでは、レコードごとに種々の項目（この例では、「性別」、「年齢」、「身長」および「体重」）に値が与えられている。本実施の形態にかかる情報処理装置では

、これら表形式データを、原理的には、図4に示すようなデータ形式に基づいて保持する。

【0040】

図4に示すように、順序集合の配列OrdSetには、順序番号ごとにレコード番号が値として配置される。この例では、すべてのレコードが表されるため、順序番号とレコード番号とは一致する。

【0041】

たとえば、性別に関しては、実際の項目値である「男」或いは「女」という値が、所定の順序にてソートされた値リストVLと、順序集合の配列OrdSet中の要素（レコード番号）のそれぞれに対応して、当該レコード番号が指示する値リスト中の番号が格納された、値リストへのポインタ配列VNoとにより、表形式データを表す。この値リストVLおよびポインタ配列VNoの組み合わせを「情報ブロック」とも称する（性別に関する情報ブロックは符号401に対応する）。

【0042】

順序集合の配列OrdSet中の要素（レコード番号）が指示する位置にある、ポインタ配列VNo中の値を特定し、さらに、その値が指示する位置にある値リストVL中の項目値を取り出すことにより、レコード番号に対応する項目値を取得することができる。他の項目の情報ブロックについても同様の構造である。

【0043】

単一のコンピュータが、単一のメモリ（物理的には複数であっても良いが、単一のアドレス空間に配置されアクセスされるという意味で単一のメモリ）であれば、当該メモリに、順序集合の配列OrdSet、各情報ブロックを構成する値リストVLおよびポインタ配列VNoとを記憶しておけばよい。しかしながら、大量のレコードを保持するためには、その大きさにともなってメモリ容量も大きくなるため、これらを分散配置できるのが望ましい。また、処理の並列化の観点からも、分散配置された情報を分掌把握できるのが望ましい。

そこで、本実施の形態においては、複数のPMMが、重なることなくレコードのデータを分掌把握し、PMM同士のパケット通信により、高速な検索、クロス集計、検索を実現している。

【0044】**[コンパイル処理]**

まず、複数のPMMにデータを分散配置し、かつ、これらを利用可能にするための処理（コンパイル処理）について説明する。たとえば、図5に示すように、4つのPMM（PMM-0～PMM-3）に、所定のレコード数のデータを収容することを考える。この例では、レコード番号0～2に関する一連のデータ、レコード番号3、4に関する一連のデータ、レコード番号5～7に関する一連のデータ、および、レコード番号8、9に関する一連のデータを、それぞれ記憶することとした。各PMMにおいても、上記表形式データの部分は、情報ブロックの形式で記憶される。

【0045】

図6および図7は、初期的にPMM-0～4の各々にてそれぞれ分掌される表形式データの例を示す図である。これらの図から、各PMMには、項目ごとの情報ブロックの部分集合などが収容される。たとえば、図6において、項目「性別」の情報ブロック601では、もとのポインタ配列VNo（図4参照）の部分集合VNo（これも「ポインタ配列」と称する。）と、もとの値リストVL（図4参照）の部分集合VL（これも「値リスト」と称する。）とが含まれる。

【0046】

ポインタ配列VNoの要素の数は、PMMが分掌するレコードの数に一致する。これに対して、値リストVLは、ポインタ配列VNoが示す値のみが抽出される。項目「性別」については、ポインタ配列VNoの値が、値リストVL全ての要素（項目値）を指示しているため、値リストVLと、もとの値リストVLとは一致する。その一方、項目「年齢」、「身長」および「体重」については、もとの値リストVLから、ポインタ配列中の要素が指示する値のみが、もとの値リストVLの部分集合として取り出されることが理解できるであろう。

【0047】

さらに、分掌される情報ブロックにおいては、各PMMにおいて、ポインタ配列VNoの要素により適切に値リストVLの要素（項目値）が指示されるように、つまり、PMM内のローカルな処理（ポインタ値の指定や項目値の指定）において

ても整合性が保たれるように、その要素は、対応するもとのポインタ配列VNoの要素から変換されている。

【0048】

前述したように、分掌される情報ブロックにおいては、値リストVLにおいて、当該分掌された情報ブロックにおいて必要な要素（項目値）のみを保持している。よって、ポインタ配列VNoおよび値リストVLによって、ローカルな処理の整合性は保たれる。しかしながら、PMM間での処理の整合性を保つため、各PMMにて分掌される値リストVLの要素（項目値）の、値リスト全体における位置づけ、つまり、各項目値が、値リスト全体において、所定の順序のもと何番目であるかを把握する必要がある。そこで、本実施の形態では、分掌される各情報ブロックにおいて、グローバル値番号配列GVNoを配置し、項目値に対応する値の位置を示す番号を収容できるようにしている。

【0049】

各PMMには、上記情報ブロックの部分集合を分掌するためのオフセット値(OFFSET)が割り当てられる。このオフセット値OFFSETは、PMMが分掌するレコードに関するもとの順序集合OrdSetにおける先頭の値に対応する。

また、各PMMにおいては、ローカルな処理における整合性をたもつため、新たな順序集合OrdSetが作られる。順序集合OrdSetの要素の数は、PMMが分掌するレコード数と一致する。その一方、PMM間での処理の整合性を保つため、各PMMが分掌するレコードが、全体の中ではどういった番号（順序集合の要素）を持っているかを把握しておく必要がある。このため、全体における各レコードの番号を収容したグローバル順序集合配列GOrdを設けている。

【0050】

図8は、本実施の形態にかかるコンパイル処理を概略的に示すフローチャートである。図8に示すように、まず、各PMMに、図6～図7に示す初期的な情報ブロックが生成される（ステップ801）。これは、たとえば、外部の他のコンピュータから、PMMに、それぞれが分掌すべき、順序集合OrdSet、各情報ブロックを構成するポインタ配列VNo、値リストVL、および、オフセット値OFFSETが与えられることにより実現できる。これら配列は、各PMM内のメモリ24に記

憶される。

【0051】

ステップ802以降は、各PMMにおけるローカルな処理およびPMM間のパケット通信にかかる処理に移行する。各PMMの制御回路20は、オフセット値を参照して、グローバル順序集合配列G0rd中に配置するそれぞれの値を算出し、配列中に値を配置する（ステップ802）。図9は、図6～図7に示す例でのグローバル順序集合配列G0rdへの値の配置を示す図である。ここでは、順序集合の値にオフセット値OFFSETを加えたものを、グローバル順序集合配列G0rdの対応する位置に配置すればよい。ステップ1002は、各PMMにおけるローカルな処理で実現できる。

【0052】

次いで、グローバル値リスト番号配列GVNoの値が決定される（ステップ803）。より詳細には、まず、各PMMにおいて、グローバル値番号配列の各々の要素として、昇順で初期値が与えられる（ステップ811）。図10、図11に示すように、たとえば、3つの要素を有するPMM-0では、先頭から昇順で、0、1および2の値が初期値となる（符号1001参照）。他のPMM（PMM-1～3）についても、並列的にグローバル値リスト番号配列GVNoの初期値が設定される（符号1011、1101、1111参照）。

【0053】

次いで、隣接するPMM間で対を形成し、対を形成する一方のPMMが、当該対となったPMM間を接続する一方のバス（たとえば、第1のバス14）を利用して、自己の値リストVLに含まれる一連の要素（項目値）をパケット化して送出し、その一方、他方のPMMが、他方のバス（たとえば、第2のバス16）を利用して、自己の値リストVLに含まれる一連の要素（項目値）をパケット化して送出する（ステップ812）。図10の例では、PMM-0からPMM-1宛てに、パケット[18,21,24]が送出され、PMM-1からPMM-0宛てに、パケット[16,28]が送出される。また、図11の例では、PMM-2、PMM-3から、パケット[16,20,33]、[18,24]がそれぞれ送出される。

【0054】

各PMMは、受理したパケットを参照して、パケット中の値（他のPMMの項目値）と自己の値リストVL中の項目値とを比較し、当該他のPMMの項目値を考慮した相対的な値の位置（順位）を特定する（ステップ813）。この位置（順位）にしたがって、グローバル値番号配列GVNoの値が更新される（ステップ814）。なお、PMMは、更新とともに、自己のPMM、および、対をなす他のPMMの値リスト中の項目値の総数を、一時的に記憶しておく（ステップ815）。

【0055】

図10の例において、PMM-0は、パケット[16,28]を受理し、自己のVL[18,21,24]と比較する。ここで、「16<18<21<24<28」であるため、グローバル値番号配列GVNoの値が、それぞれ、「1」、「2」および「3」に更新される（符号1021参照）。その一方、PMM-1は、パケット[18,21,24]を受理し、自己のVL[16,28]と比較する。その結果、グローバル値番号配列GVNoの値が、それぞれ、「0」および「4」に更新される（符号1031参照）。図11に示すように、PMM-2およびPMM-3の間でも同様の処理が実行される。なお、PMMは、受理したパケットを次のタイミングでパケットが送られた方向に送出すればよい。このパケットは受け取り手がないため廃棄される。

【0056】

次いで、先に形成された対のPMMをPMM群として、隣接するPMM群の対が形成され。なお、PMM数が2のべき乗であれば、PMM群の対を形成できるが、そうでない場合には、PMM群の対を形成できない部分については、そのままにしておけばよい。たとえば、PMMの数が3であれば、PMM-0およびPMM-1からなるPMM群と、PMM-2との対を形成すれば良い。

【0057】

その後、一方のPMM群における時計回りの方向の上流側のPMMから、時計回りでバス（第1のバス）上にパケットが送出される一方、他方のPMM群における時計回り方向の下流側のPMMから、反時計回りでバス（第2のバス）上にパケットが送出される。図10、図11に示した例に関しては、PMM-0から時計回りで第1のバス上にパケットが送出されるとともに、PMM-3から反時

計回りで第2のバス上にパケットが送出される（図12参照）。

【0058】

より詳細に、当初パケットを送出するPMMは、ステップ815で記憶した項目値の数に基づき、値の個数が項目値の数に一致し、かつ、PMM自身の値リストVLの要素（項目値）を、対応するグローバル値番号配列GVNoの値が示す位置に配置し、その一方、他の位置にはNULL値を配置したようなパケットを生成して（ステップ1401～1403）、所定の方向に、次のPMM宛で送出する（ステップ1404）。

【0059】

図12に示すように、たとえば、PMM-0においては、PMM-0およびPMM-1の対で、5つの項目値があることがわかっている。したがって、5つの値を持つパケット [-, 18, 21, 24, -]（ここに、「-」はNULL値を表す）が、時計回りに第1のバスを介してPMM-1に送出される。その一方、PMM-3においては、PMM-2およびPMM-3の対で、5つの項目値があることがわかっている。したがって、5つの値を持つパケット [-, 18, -, 24, -] が、反時計回りに第2のバスを介してPMM-2に送出される。

【0060】

パケットの宛先として、当該パケットを受理したPMMは、自己のグローバル値番号配列GVNoを参照して、パケット中のNULL値の所定の位置、つまり、グローバル値番号配列GVNoの要素が示す位置に、対応する項目値を配置する（ステップ1411、1412）。その後、PMMは、パケットが流れてきた方向に沿って、次のPMMにパケットを送出する（ステップ1413）。

【0061】

図12において、PMM-0からのパケットを受理したPMM-1は、パケット中、NULL値が配置された位置のうち、自己のグローバル値番号配列GVNoの要素「0」および「4」が示す位置に、対応する値「16」および「28」をそれぞれ配置する。これにより、パケット[16, 18, 21, 24, 28]が、第1のバスを介して時計回りに送出される。また、PMM-3からのパケットを受理したPMM-2も、同様に、パケット[16, 18, 20, 24, 33]を、第2のバスを介して反時計回りに送出する。

上記例においては、一群のPMMが、2つのPMMにて構成されているが、一群のPMMが、3つ以上のPMMにて構成されている場合には、さらに、パケットを受理したPMMにおいて、図14（b）に示す処理が実行される。

【0062】

次に、一群のPMMと対をなす他のPMMの一群を構成するPMMによりパケットが受理された際に実行される処理について説明する。

図15（a）に示すように、他のPMMの一群を構成するPMMが、パケットを受理すると（ステップ1501）、当該PMMは、自己が送出したパケットと、受理したパケットとを比較して（ステップ1502）、受理したパケット中、自己が送出したパケットと同一の項目値、つまり、重複する項目値があれば、これを消去する（ステップ1503）。次いで、PMMは、重複する項目値が消去されたパケットを参照して、パケット中の値（他のPMMの一群の項目値）と自己の値リスト中の項目値とを比較し、他のPMMの一群の項目値を考慮した相対的な値の位置（順位）を特定する（ステップ1504）。この位置（順位）にしたがって、グローバル値番号配列GVNoの値が更新される（ステップ1505）。

【0063】

重複する項目値が消去された後のパケットは、受理した方向と同一の方向に、それぞれ、バスを介して、同一のPMMの一群を構成する隣接するPMMに送出される（ステップ1506）。

【0064】

図13において、PMM-1は、反時計回りに、隣のPMM群を構成するPMMであるPMM-2から、パケット[16, 18, 20, 24, 33]を受理する。その一方、PMM-1は、時計回りに、PMM-2に対して、パケット[16, 28, 21, 24, 28]を送出している。そこで、PMM-1は、両者を比較して、受理したパケットのうち、「16」、「18」および「24」が重複しているため、これらを削除して、パケット[20, 33]に更新する（符号1300参照）。その上で、そのパケット中の項目値「20」、「33」を考慮して、「16<20<28<33」であることから、自己のグローバル値番号配列GVNo中、項目値「16」に対応する値「0」は変更されず、項目値「28」に対応する値「4」は「5」に更新される（符号1301参照）。

また、パケット [20,33] が、反時計回りにPMM-0宛てに送出される。

【0065】

これに対して、PMM-2は、時計回りに、隣のPMM群を構成するPMMであるPMM-1から、パケット [16,18,21,24,28] を受理している。PMM-2においても、受理したパケットと、送出したパケットを比較して、受理したパケット中の値のうち、重複する値を削除して、パケット [21,28] に更新する（符号1300参照）。次いで、PMM-2は、パケット中の項目値を考慮して、「16<20<21<28<33」であることから、自己のグローバル値番号配列GVNo中、項目値「16」および「20」にそれぞれ対応する値「0」、「2」を更新せず、その一方、項目値「33」に対応する値「4」を「6」に更新する（符号1302参照）。また、パケット [21,28] が、時計回りにPMM-3宛てに送出される。

【0066】

同一のPMM群を構成するPMMからパケットを受理したPMMは（ステップ1511参照）、受理したパケット中の値と自己の値リストVL中の項目値との比較、および、自己の項目値の相対的な値の位置（順位）の特定（ステップ1512）、並びに、特定された位置（順位）に基づくグローバル値番号配列GVNoの更新（ステップ1513）を実行する。また、パケットは、送られてきた方向と同一の方向に次のPMMに向けて送出される（ステップ1514）。なお、次のPMMが存在しなければ、そのパケットは廃棄される。その一方、次のPMMが存在すれば、当該PMMにおいて、ステップ1511～1513の処理が実行される。

【0067】

図13において、符号1310には、送られてきたパケット [20,33] を考慮したグローバル値番号配列GVNoの更新が示され、符号1313には、送られてきたパケット [21,28] を考慮したグローバル値番号配列GVNoの更新が示されている。

このようにして、PMM間の処理の整合をはかるためのグローバル順序集合配列GOrdやグローバル値番号配列GVNoが生成されることにより、コンパイル処理が完了する。なお、グローバル値番号配列については、項目ごとに処理が実行され

、項目ごとのグローバル値番号配列が得られる。

【0068】

コンパイル処理が終了すると、検索、クロス集計、ソートなどの処理を円滑かつ迅速に実行することができる。

【0069】

[検索処理]

次に、検索処理について説明する。図16に示すように、まず、各PMMはて、検索対象となった項目について、値リストVLと同じサイズのフラグ配列を作成し（ステップ1601）、次いで、フラグ配列の値を合否条件でセットアップする（ステップ1602）。このセットアップに際して、検索条件に合致する項目値に対応するフラグ配列の値として「1」を設定し、それ以外のフラグ配列の値として「0」を設定する。

【0070】

次いで、各PMMは、検索結果格納先領域である新たなグローバル順序集合配列GOrd' およびOrdSet' とを生成する（ステップ1603）。図17は、各PMにおいて、値がセットアップされたフラグ配列および領域として新たなグローバル順序集合配列および順序集合配列が生成された状態の一例を示す図である。この例では、「年齢」という項目について、「20歳以上24歳以下」のレコードを検索することとしている。したがって、各PMMにおいて、項目値が20以上24以下であるようなものに対応するフラグ配列の値が「1」となっている。

【0071】

次に、合否判定実行される（ステップ1604）。この処理においては、順序集合配列OrdSetの値ごとに、値リストへのポインタVNoの値（ポインタ値）が見出され、当該ポインタ値が示すフラグ配列の値を取得する（ステップ1611）。この値が「0」であれば（ステップ1612でノー（No））、なんら処理を実行しない。その一方、フラグ配列の値が「1」であれば（ステップ1612でイエス（Yes））、新たなグローバル順序集合配列GOrd' および順序集合配列OrdSet' に、順次、処理に関連するグローバル順序集合配列GOrdおよび順序集合配列OrdS etの値が、それぞれ収容される（ステップ1613）。

【0072】

順序集合配列の末尾の要素まで、ステップ1611～1613の処理が繰り返される（ステップ1614、1615参照）。上記図16の処理は、各PMMにおいてローカルに、かつ、並列的に実行される。図18は、各PMMにおいて、図16の処理が実行され、ローカルにかつ並列的に、新たなグローバル順序集合配列G0rd' および順序集合配列0rdSet' に値が配置される状態の例を示す。また、図19は、配列中、不要な領域を削除した状態を示す（符号1901～1904参照）。

【0073】

上記処理の後、PMM間のパケット通信にかかる処理に移行する。まず、各PMMは、検索条件に合致するレコードのPMM全体における位置（順位）を格納するための新たなグローバル順序集合配列G0rd'' を生成し（ステップ2001）、配列に昇順で初期値を格納しておく（ステップ2002）。この新たなグローバル順序配列集合G0rd'' のサイズは、配列G0rd' やG0rdのサイズと一致する。

【0074】

次いで、隣接するPMM間で対を形成し、対を形成する一方のPMMが、当該対となったPMM間を接続する一方のバス（たとえば、第1のバス14）を利用して、先に図16に示す処理にて生成された配列G0rd' の要素をパケット化して、対をなす他のPMMに送出する（ステップ2003）。その一方、他のPMMも、同様に、対となったPMMを接続する他方のバス（たとえば、第2のバス16）を利用して、配列G0rd' の要素をパケット化して、対をなすPMMに送出する。

【0075】

図21の例において、PMM-0からは、G0rd' に対応するパケット [1,2] がPMM-1宛てに送出され、PMM-1からは、G0rd' に対応するパケット [Φ]（空集合）が送出される。PMM-2とPMM-3との間でも、同様にパケットの授受が行なわれる。

【0076】

パケットを受理すると、PMMは、受理したパケットを参照して、パケット中

の値と、自己の配列GOrd' 中の値とを比較し、他のPMMの配列GOrd' を考慮した値の位置（順位）を特定する。（ステップ2004）。この位置にしたがって、新たなグローバル順序集合配列GOrd' ' の値が更新される（ステップ2005）。これにより、対のPMMにおける項目値の順位が確定する。パケットは、それぞれ、受理した方向と同一の方向に送出されるが、ここでは受け取り手がないため廃棄される。なお、PMMは、自己のPMMおよび対をなす他のPMMにおける配列GOrd' の要素の総数を、一時的に記憶しておく（ステップ2006）。

【0077】

図21の例では、PMM-0は、PMM-1から与えられたパケット（要素は空集合）を参照する。パケットの要素が空集合であるため、配列GOrd' ' の値は変化しない。PMM-1においても、そもそも配列の要素が空であるため、なんら処理は実行されない。

【0078】

その一方、PMM-2は、PMM-3から与えられたパケット[8]を参照して、自己の配列GOrd' と比較する。ここに、「8>5」であるため、GOrd' ' の値は変化しない。その一方、PMM-3においては、受理したパケット[5]と自己のGord' の値とが比較される。ここでは、「5<8」であるため、GOrd' ' の値が、「0」から「1」に更新される。

【0079】

次いで、先に形成された対のPMMをPMM群として、隣接するPMM群の対が形成される。これについては、コンパイル処理の場合と同様である。その後、一方のPMM群における時計回りの方向の上流側のPMMから、時計回りでバス（第1のバス）上にパケットが送出される一方、他方のPMM群における時計回り方向の下流側のPMMから、反時計回りでバス（第2のバス）上にパケットが送出される。図22においては、PMM-0から時計回りで第1のバス上にパケットが送出されるとともに、PMM-3から反時計回りで第2のバス上にパケットが送出される。

【0080】

より詳細に、当初パケットを送出するPMMは、ステップ2005で記憶した

配列G0rd' の総数に基づき、値の個数が当該総数に一致し、かつ、PMM自身の配列G0rd' の値を、対応する配列G0rd' ' の値が示す位置に配置し、その一方、他の位置にはNULL値を配置したようなパケットを生成して（ステップ2201～2203）、所定の方向に、次のPMM宛で送出する（ステップ2204）。

【0081】

図23に示すように、たとえば、PMM-0においては、PMM-0およびPMM-1の次いで、2つの項目値があることがわかっている。したがって、2つの値を持つパケット[1,2]が、時計回りに第1のバスを介してPMM-1に送出される。その一方、PMM-3においては、PMM-2およびPMM-3の次いで、2つの項目値があることがわかっている。したがって、2つの値を持つパケット[-,8]（ここに、「-」はNULL値を表す）が、反時計回りに第2のバスを介してPMM-2に送出される。

【0082】

パケットの宛先として、当該パケットを受理したPMMは、自己の配列G0rd' ' を参照して、パケット中のNULL値の所定の位置、つまり、配列G0rd' ' の値が示す位置に、対応する配列G0rdの値を配置する（ステップ2211、2212）。その後、PMMは、パケットが流れてきた方向に沿って、次のPMMにパケットを送出する（ステップ2213）。

【0083】

図23において、PMM-0からのパケットを受理したPMM-1は、G0rd' ' に要素が存在しない（つまり、「空」である）ため、パケットをそのまま、第1のバスを介して時計回りに送出する。また、PMM-3からのパケットを受理したPMM-2は、パケットの所定の位置に、自己のG0rd' の要素を配置し、パケット[5,8]を、第2のバスを介して反時計回りに送出する。

上記例においては、一群のPMMが、2つのPMMにて構成されているが、一群のPMMが、3つ以上のPMMにて構成されている場合には、さらに、パケットを受理したPMMにおいて、図22（b）に示す処理が実行される。

【0084】

次に、一群のPMMと対をなす他のPMMの一群を構成するPMMによりパケ

ットが受理された際に実行される処理について説明する。

図23に示すように、自己に宛てられたパケットを受理したPMMは（ステップ2401参照）、受理したパケット中の値と、自己の配列G0rd' 中の値とを比較して、配列G0rd' 中の値の相対的な位置（順位）を特定する（ステップ2402）。次いで、PMMは、特定された順位に基づいて、配列G0rd' ' を更新する（ステップ2403）。その後、PMMは、パケットを、送られてきた方向と同一の方向に沿って、次のPMMに向けて送出する（ステップ2404）。なお、次のPMMが存在しなければ、そのパケットは廃棄される。その一方、次のPMMが存在すれば、当該PMMにおいて、ステップ2401～2404の処理が実行される。

【0085】

図25の例において、PMM-2からのパケット [5,8] を受理したPMM-1は、自己の配列G0rd' に要素を持っていないため、パケットをそのまま同じ方向に沿って、PMMの0宛に送出する。また、PMM-1からのパケット [1,2] を受理したPMM-2は、上記パケットの値と、配列G0rd' の値とを比較する。ここでは、「1<2<5」であるため、G0rd' ' の値が、「0」から「2」に更新される。さらに、パケットはPMM-3に送出される。

【0086】

PMM-1からパケットを受理したPMM-0でも、受理したパケットと、自己のG0rd' の値とを比較する。ここでは、「1<2<5<8」であるため、G0rd' ' の値は変化しない。その一方、PMM-2からのパケットを受理したPMM-3における受理したパケットと、自己のG0rd' の比較においては、「1<2<8」であるため、G0rd" の値は、「1」から「3」に更新される。

【0087】

このような処理が並列的に実行されることにより、PMMの配列G0rd' ' の値が確定する。配列G0rd' ' の値は、検索により抽出されたレコードの、全体における順位、つまり、グローバルな順位を表す。G0rd' ' を新たなG0rdとすれば、当該配列G0rd中の値にしたがって、順次レコードを取り出せば、所定の順序にしたがった検索結果を取得することが可能となる。

【0088】

図26における各PMMの配列G0rdが、それぞれ、検索処理の結果得られた新たな配列である。この配列G0rdの値の小さな順に、対応する配列OrdSetの値、この値が示す値リストVLの値を取り出せば、「年齢」という項目について「20歳以上24歳以下」のレコードが、レコード番号（順序集合）の順でリストされ得る。

【0089】

[クロス集計処理]

次に、クロス集計処理について説明する。ここでも、コンパイル処理が終了した状態から処理が開始される。図27に示すように、各PMMは、クロス集計をする項目に関する複数の値リストVLのサイズを乗じたサイズのカウントアップ領域を生成する（ステップ2701）。次いで、この領域中の各値が「0」に初期化される（ステップ2702）。図28は、「性別」および「年齢」という項目について、各PMMにおいて、（「性別」に関する値リストVLのサイズ）×（「年齢」に関する値リストVLのサイズ）というサイズを有する領域が作られ、それに初期値「0」が与えられた状態を示す。

【0090】

次いで、各PMMは、カウントアップ配列のそれぞれに対するカウントアップ処理を実行する（ステップ2703）。より詳細には、各PMMは、順序集合配列OrdSetの値を参照して、クロス集計すべき項目のそれぞれのポインタ配列VNoの値を特定する（ステップ2711）。次いで、クロス集計すべき複数のポインタ配列VNoの値にて特定される、前記領域中の位置の値がカウントアップされる（ステップ2712）。このような処理が、順序集合配列OrdSetの末尾まで繰り返される（ステップ2713、2714参照）。

【0091】

図29は、各PMMにおけるカウントアップの一例を示す図である。たとえば、PMM-0において、順序集合配列OrdSetの要素「0」が示す位置の、項目「性別」のポインタ配列VNoの値は「1」、項目「年齢」のポインタ配列VNoの値は「0」である。したがって、カウントアップ領域中の（性別、年齢） = (1, 0)

で特定される領域の値が、「0」から「1」にカウントアップされる。他のPMMにおいても、同様の処理が実行されていることが理解できるであろう。

【0092】

カウントアップが終了すると、以下の処理では、カウントアップ領域において、軸として実際の項目値の代わりに、各項目のグローバル値番号配列GVNoの値をキーに設定される。つまり、グローバル値番号配列GVNoの値を指定することにより、カウントアップ領域の値が特定されることになる。

図30は、それぞれのPMMにおいて、グローバル領域の値を特定するために、グローバル値番号配列GVNoの値を利用した状態を示す図である。実際には、このようなグローバル領域の項目として値を与えるような処理を実行するのではなく、以下の処理で、これら配列GVNoが利用されるに過ぎない。

【0093】

PMMにおいては、共通の論理座標配列を用いて、カウントアップ領域のそれぞれの値を合算する。より詳細には、図31(a)に示すように、あるPMMが、論理座標配列のパケットを受理すると(ステップ3101)、集計にかかる複数の項目の配列GVNoにより特定されるカウントアップ領域の値を、論理座標配列中、上記複数の項目の配列GVNoに割り当てられた位置の値に加える(ステップ3102)。なお、最初に論理座標配列を生成し、上記値を設定する処理を実行するPMMにおいては、論理座標配列の受理の代わりに、論理座標配列の生成および初期値の配置が実行される(ステップ3100参照)。

【0094】

論理座標配列について説明する。論理座標配列は、論理座標の値とこれに対応するカウント値とから構成される。論理座標の値は、クロス集計の対象となった項目の多次元の座標と一意的に対応付けられている。たとえば、項目「性別」と項目「年齢」とでクロス集計をする場合には、(項目「性別」に関する配列GVNoの値、項目「年齢」に関する配列GVNoの値)と、論理座標の値とが一意的に対応付けられている。この対応付けは、予め各PMMに通知され、各PMMにて記憶されている。図32は、初期的に論理座標配列を生成したPMM-0における論理座標配列におけるカウントアップの例を示す図である。この例では、論理座標

と、多次元の座標とが以下のように対応付けられている。

【0095】

0 = (項目「性別」のGVNoの値「0」, 項目「年齢」のGVNoの値「0」)

1 = (項目「性別」のGVNoの値「0」, 項目「年齢」のGVNoの値「1」)

2 = (項目「性別」のGVNoの値「0」, 項目「年齢」のGVNoの値「2」)

3 = (項目「性別」のGVNoの値「0」, 項目「年齢」のGVNoの値「3」)

4 = (項目「性別」のGVNoの値「0」, 項目「年齢」のGVNoの値「4」)

:

12 = (項目「性別」のGVNoの値「1」, 項目「年齢」のGVNoの値「5」)

13 = (項目「性別」のGVNoの値「1」, 項目「年齢」のGVNoの値「6」)

【0096】

PMM-0は、座標(0,3)、(1,1)および(4,1)の値が「1」にセットされているため、論理座標が「3」、「8」および「11」に対応するカウントアップ領域の値を、それぞれ、セットされた値だけカウントアップする。

【0097】

このような処理の後、PMMは、前記論理座標配列をパケット化し（ステップ3103）、隣接する所定のPMMに宛てて送出する（ステップ3104）。図32の例では、パケットが、PMM-1宛てに送出される。

受理したPMMは、同様に、図30の処理を実行して、先にパケットが送られた方向に沿って、隣接するPMM宛に、パケットを送出する。図32ないし図35は、それぞれ、PMM-1ないしPMM-3にて、順次実行される論理座標配列のカウントアップ処理を示す図である。

【0098】

このようにして、各PMMに論理座標配列が与えられ、それぞれのカウントアップ領域の値に応じて、論理座標配列の値が確定する。最終のPMM（例では、PMM-3）からは、再度、初期的に論理座標配列を生成したPMMに、値が確定した論理座標配列のパケットが与えられる。図31(b)に示すように、PMMが、再度、論理座標配列のパケットを受理すると（ステップ3111）、カウントアップ領域の値を、それぞれ、対応する論理座標配列中の値に更新する（ス

ステップ3112）。つまり、図31（a）の処理では、カウントアップ領域の値を論理座標配列中に書き込む形態であったが、図31（b）の処理では、論理座標配列の値を、カウントアップ配列中に書き込む。ここでは、図31（a）の処理と同様に、カウントアップ領域の各値を特定する座標と、論理座標配列との対応関係が利用される。

【0099】

値の更新が終了すると、PMMは、パケットが送られてきた方向に沿って、隣接する所定のPMMに、当該パケットを送出する（ステップ3113）。図36ないし図37は、PMM-0ないしPMM-3において、それぞれ順次実行される、論理座標配列の値の取り込み処理（図35の処理）を示す図である。

【0100】

このようにして、図40に示すように、各PMMにおいて、クロス集計の対象となった項目の多次元の座標ごと、当該項目の項目値の組み合わせに関する集計値を得ることができる。図40において、符号4001ないし4004は、それぞれ、PMM-0ないしPMM-3に関する集計結果を示すものである。これら集計結果の配列（カウントアップ領域）は、PMMのメモリに記憶される。

【0101】

【ソート処理】

次に、ソート処理について説明する。ここでも、コンパイル処理が終了した状態から処理が開始される。図41に示すように、各PMMは、ソートすべき項目に関する値リストVLと同一のサイズの、存在数配列の領域を生成し（ステップ4101）、領域中の各値に初期値「0」を与える（ステップ4102）。図42は、「年齢」という項目について、それぞれのPMMにおいて、値リストVLと同一のサイズを有する領域が作られ、それぞれに初期値「0」が与えられた状態を示す。

【0102】

次いで、各PMMは、存在数配列のそれぞれに対するカウントアップ処理を実行する（ステップ4103）。より詳細には、各PMMは、順序集合配列OrdSetの値を参照して、ソートすべき項目のポインタ配列VNoの値を特定する（ステッ

4111)。次いで、各PMMは、存在数配列中、当該ポインタ配列VNoの値に示される位置の値をカウントアップする(ステップ4112)。このような処理が、順序集合配列OrdSetの末尾まで繰り返される(ステップ4113, 4114参照)。

【0103】

図43は、各PMMにおけるカウントアップの一例を示す図である。たとえば、PMM-0において、順序集合配列OrdSetの要素「0」が示す位置の、年齢のポインタ配列VNoの値は「0」である。したがって、存在数配列の「第0番目」の位置、つまり、先頭の位置にある値を、「0」から「1」にカウントアップする。他のPMMにおいても、同様の処理が実行されていることが理解できるであろう。

【0104】

カウントアップ処理が終了すると、図44に示すように、各PMMは、存在数配列の要素を累計して、当該存在数配列を累計数配列に変換する(ステップ4401)。累計数配列の要素である累計数は、項目値を指示するレコードの数を示す存在数を考慮して、当該累計数が配置されている位置の項目値を指示するレコードの先頭の位置を示すようになっている。具体的には、各PMMが、配列の位置を示すパラメータ「i」を初期化して(ステップ4411)、パラメータが示す存在数配列中の値を取り出し(ステップ4412)、パラメータ「i」が示す位置より、後ろの位置、つまり、「i+1」、「i+2」、・・・の位置の存在数配列の値に、ステップ4412で取り出された値を、それぞれ加算する(ステップ4413)。ステップ4412、4413に示す処理は、値リストVLの要素(項目値)の個数だけ繰り返せばよい(ステップ4414、4415参照)。

【0105】

このようにして、たとえば、図45に示すような累計数配列を得ることができる。さらに、各PMMは、後にPMM全体における順位を格納するための配列GVNo、GOrd'およびOrdSet'のための領域も作られる(ステップ4402)。これら配列のサイズは、それぞれ、値リストVLのサイズと一致する。

【0106】

次に、各PMMにおけるローカルなソート処理が実行される。図46に示すように、各PMMは、順序集合配列OrdSetの値を取り出し（ステップ4601）、次いで、ポインタ配列VNo中、配列OrdSetの値が指示する位置の値（ポインタ値）を特定する（ステップ4602）。その後、各PMMは、ソートすべき項目のグローバル値番号配列GVNo中、ポインタ配列VNoの値が示す位置の値を取得する（ステップ4603）。この値は、後述する値の格納処理に利用される。その一方、上記累計数配列においても、ポインタ配列VNoが示す位置の値が取得される（ステップ4604）。この値は、後述する値の格納処理において、配列中の位置を指定するために利用される。

【0107】

次に値の格納処理が実行される。各PMMは、先に生成した配列GVNo中、ステップ4604で取得された累計数配列の値が示す位置に、ステップ4602で取得された、ソートすべき項目に関するGVNoの値を配置する（ステップ4605）。また、各PMMは、配列GOrd'、OrdSet'中、ステップ4604で取得された累計数配列の値が示す位置に、グローバル順序集合配列GOrdおよび順序集合配列OrdSetの値を、それぞれ配置する（ステップ4606）。次いで、処理に用いられた累計数配列の値がインクリメントされる（ステップ4607）。

【0108】

上記ステップ4601～4607の処理が、配列OrdSet中の全ての値について、順次実行される（ステップ4608、4690参照）。

図47および図48は、各PMMにおいてローカルなソート処理が実行されている状態の例を示す図である。たとえば、PMM-0に関して、図47においては、配列OrdSetの値「0」の取り出し（ステップ4601参照）、当該OrdSetの値「0」が示す位置の、配列VNoの値「0」の特定（ステップ4602参照）、当該配列VNoの値「0」が示す位置の、配列GVNoの値「1」の取得（ステップ4603）、および、配列VNoの値「0」が示す位置の、累計数配列の値「0」の取得（ステップ4603）が実行されていることが理解できるであろう。また、累計数配列の取得の後、当該累計数配列の値が、「0」から「1」になっていることもわかる（ステップ4607参照）。

【0109】

また、PMM-0に関して、図48において、ステップ4603で取得された累計数配列の値の示す位置における、配列GVNo、G0rd' および0rdSet'への、項目「年齢」に関する配列GVNoの値「1」、並びに、配列G0rdの値「0」および配列0rdSetの値「0」の配置（ステップ4605、4606）が示されていることが理解できるであろう。他のPMMについても、図47、48において、同様にステップ4601～4605に示す処理が実行されていることがわかる。

【0110】

このような各PMMにおけるローカルなソート処理が終了すると、PMM間のパケット通信により、PMM全体のソート処理が実行される。図49に示すように、各PMMにおいて、PMM全体におけるソート順を格納するための配列G0rd' の領域が生成され（ステップ4901）、各配列において昇順で初期値が与えられる（ステップ4902）。これら処理により、図50に示すように、各PMMにおいて、初期値が配置された配列G0rd' が作られる。

次いで、隣接するPMM間で対を形成し、対を形成する一方のPMMが、当該ついになったPMM間を接続する一方のバス（たとえば、第1のバス14）を利用して、自己の有する配列GVNoおよびG0rd' の値の組をパケット化して送出する（ステップ4903）。同様に、他方のPMMも、他方のバス（たとえば、第2のバス16）を利用し、自己の有する配列GVNoおよびG0rd' の組をパケット化して送出する（ステップ4903）。

【0111】

図51の例では、PMM-0からPMM-1宛てに、パケット $[(1, 0), (3, 1), (4, 2)]$ が送出され、PMM-1からPMM-0宛てに、パケット $[(0, 3), (5, 4)]$ が送出される。同様に、PMM-2からPMM-3宛てに、パケット $[(0, 6), (2, 5), (6, 7)]$ が送出され、PMM-3からPMM-2宛てに、パケット $[(1, 9), (4, 8)]$ が送出される。

パケットを受理したPMMは、受理したパケット中、対をなす他のPMMの配列GVNoの値と、自己の配列GVNoの値とを比較して、当該他のPMMの配列GVNoを考慮した、相対的な値の位置（順位）を特定する（ステップ4904）。この位

置（順位）にしたがって、配列G0rd'，の値が更新される（ステップ4905）。なお、PMMは、更新とともに、自己のPMM、および、対をなす他のPMMの配列GVNo中の値の総数を、一時的に記憶しておく（ステップ4906）。

【0112】

図51の例において、PMM-0は、パケット $[(0, 3), (5, 4)]$ を受理する。このパケット中、配列GVNoの値に相当する「0」、「5」と、自己の配列GVNoの値「1」、「3」、「4」とが比較される。ここで、「0<1<3<4<5」であるため、配列G0rd'，の値は、それぞれ、「1」、「2」および「3」に更新される。その一方、PMM-1は、パケット $[(1, 0), (3, 1), (4, 2)]$ を受理する。このパケット中、配列GVNoの値に相当する「1」、「3」、「4」と、自己の配列GVNoの値「0」、「5」とが比較される。ここでも、「0<1<3<4<5」であるため、配列G0rd'，の値が、それぞれ、「0」、「4」に変更される。PMM-2およびPMM-3においても、同様の処理により、それぞれG0rd'，の値が更新される。なお、PMMは、受理したパケットを次のタイミングでパケットが送られた方向に送出すればよい。このパケットは受け取り手がないため廃棄される。

【0113】

次いで、先に形成された対のPMMをPMM群として、隣接するPMM群の対が形成され。なお、PMM数が2のべき乗であれば、PMM群の対を形成できるが、そうでない場合には、PMM群の対を形成できない部分については、そのままにしておけばよい。たとえば、PMMの数が3であれば、PMM-0およびPMM-1からなるPMM群と、PMM-2との対を形成すれば良い。

【0114】

その後、一方のPMM群における時計回りの方向の上流側のPMMから、時計回りでバス（第1のバス）上にパケットが送出される一方、他方のPMM群における時計回り方向の下流側のPMMから、反時計回りでバス（第2のバス）上にパケットが送出される。図50、図51に示した例に関しては、PMM-0から時計回りで第1のバス上にパケットが送出されるとともに、PMM-3から反時計回りで第2のバス上にパケットが送出される（図53参照）。

【0115】

より詳細には、図52(a)に示すように、当初パケットを送出するPMMは、ステップ4906で記憶したPMM群全体の配列GVNoの値の総数に基づき、値の組の個数が、上記配列GVNoの総数に一致し、かつ、PMM自身の配列GVNoの値および配列G0rd'の値の組を、配列G0rd'の中の対応する値が示す位置に配置し、その一方、他の位置には、NULL値を配置したようなパケットを生成する(ステップ5201～5203)。生成されたパケットは、所定の方向に、次のPMM宛で送出される(ステップ5204)。

【0116】

図53に示すように、たとえば、PMM-0においては、PMM-0およびPMM-1の対で、配列GVNoに5つの値があることがわかっている。したがって、5つの値の組を持つパケット $[-, (1, 0), (3, 1), (4, 2), -]$ (ここに、「-」はNULL値を表す)が、時計回りに第1のバスを介してPMM-1に送出される。その一方、PMM-3においては、PMM-2およびPMM-3の対で、配列GVNoに5つの値があることがわかっている。したがって、5つの値を持つパケット $[-, (1, 9), -, (4, 8), -]$ が、反時計回りに第2のバスを介してPMM-2に送出される。

【0117】

図52(b)に示すように、パケットの宛先として、当該パケットを受理したPMMは、自己の配列G0rd'を参照して、パケット中のNULL値の所定の位置、つまり、配列G0rd'の値が示す位置に、対応する配列GVNoの値およびG0rd'の値の組を配置する(ステップ5211、5212)。その後、PMMは、パケットが流れてきた方向に沿って、次のPMMにパケットを送出する(ステップ5213)。

【0118】

図53において、PMM-0からのパケットを受理したPMM-1は、受理したパケット中、NULL値が配置された位置のうち、自己の配列G0rd'の要素「0」および「4」が示す位置に、対応する値の組(0, 3)および(5, 4)をそれぞれ配置する。これにより、パケット $[(0, 3), (1, 0), (3, 1), (4, 2), (5, 4)]$ が、第1のバスを介して時計回りに送出される。また、PMM-3からのパケットを受理したP

MM-2も、同様に、パケット $[(0, 6), (1, 9), (2, 5), (4, 8), (6, 7)]$ を、第2のバスを介して反時計回りに送出する。

【0119】

次に、一群のPMMと対をなす他のPMMの一群を構成するPMMによりパケットが受理された際に実行される処理について説明する。

図55に示すように、他のPMMの一群を構成するPMMが、パケットを受理すると（ステップ5501）、受理したパケット中、GVNoに相当する値と、自己のGVNoの値を比較して、当該他のPMMの配列GVNoを考慮した、相対的な値の位置（順位）を特定する（ステップ5502）。この位置（順位）にしたがって、配列GOrd'の値が更新される（ステップ5503）。次いで、PMMは、パケットを、当該パケットが送られた方向に沿って、次の隣接するPMMに送出する（ステップ5504）。なお、ステップ5502において、GVNoの値が同一であった場合には、組となっている配列GOrd'の値が参照され、その値が小さいものが上位に配置される。

【0120】

パケットを受理したPMMにおいては、順次、図55に示す処理が繰り返され、ソート順を示すグローバル順序集合GOrd'が完成する。処理が終了するのに際して、生成されたGOrd'をGOrdと読み替え、かつ、OrdSet'をOrdSetと読み替えるべき。

【0121】

図54の例において、PMM-1は、PMM-2から、パケット $[(0, 6), (1, 9), (2, 5), (4, 8), (6, 7)]$ を受理する。このパケット中、配列GVNoに相当する値と、自己の配列GVNoの値とを比較すると、「0=0<1<2<4<5」となるため、配列GOrd'の値は、「0」、「4」から、それぞれ、「0」、「8」に更新される。なお、配列GVNoの値が同じときには、それぞれの対応するGOrd'の値が参照される。

【0122】

次のタイミングで、PMM-0は、PMM-1から、同じパケット $[(0, 6), (1, 9), (2, 5), (4, 8), (6, 7)]$ を受理する。このパケット中、配列GVNoに相当する値と、自己の配列GVNoの値とを比較すると、「0<1=1<2<3<4=4<6」となる。したが

つて、配列G0rd'，の値は、「1」、「2」、「3」から、それぞれ、「2」、「5」、「6」に更新される。PMM-2やPMM-3においても同様の処理が実行され、それにおいて、G0rd'，の値が更新される。

【0123】

上述したように、生成されたG0rd'，をG0rdと読み替え、かつ、OrdSet'をOrdSetと読み替えることにより、たとえば、各PMMにおいて、図56に示すような配列が取得される。ここで、配列G0rdの順に、レコードを順次取り出すことにより、ソートされた表形式データを得ることができる（図57参照）。

【0124】

[システム構成、本発明の意義]

本発明にかかる情報処理システムは、たとえば、フロントエンドとなる端末装置と、リング状のチャネルを介して接続され、端末装置からの命令を、それぞれのPMMが受理することにより、PMMにおいて、上述したコンパイル、検索、クロス集計ソートの処理が実行できる。また、各PMMはパケットを何れかのバスを利用して送出すればよく、PMM間の同期等を外部から制御する必要もない。

【0125】

また、制御装置には、上記コンパイル、検索などの繰り返し演算のためのハードウェア構成を備えたアクセラレータチップのほか、これに加えて、汎用CPUを含めても良い。汎用CPUは、端末装置からチャネルを介して伝達された命令を解釈し、アクセラレータチップに必要な指示を与えることができる。

【0126】

さらに、制御装置、特に、その中のアクセラレータチップには、順序集合配列、グローバル順序集合配列など作業に必要な種々の配列を収容するためのレジスタ群が設けられているのが望ましい。これにより、いったん、メモリからレジスタ上に処理に必要な値をロードしてしまえば、コンパイル、検索、クロス集計およびソートにかかる上述した処理演算中には、制御装置はメモリにアクセスすることなく、レジスタから値を読み出し、或いは、レジスタに値を書き込めばよい。これにより、メモリアクセスの回数を著しく減じる（演算処理前のロード、お

よび、処理結果の書き込み）ことができ、処理時間を著しく短縮することが可能となる。

【0127】

次に、本発明にて導入した配列G0rdおよび配列GVNoの意義について説明する。本発明において、グローバル順序集合配列G0rdは、各PMMが掌握するローカルな表形式データを集合させたグローバルな表形式データ中、各PMMの掌握する表形式データの各レコードの位置（順位）を示している。すなわち、本発明においては、グローバル順序集合配列G0rdおよび順序集合配列OrdSetにより、レコードの位置情報を、グローバルな成分とローカルな成分とに分離し、これにより、グローバルな表形式データを扱うことが可能となるとともに、各PMMが単独で処理を実行することも可能となる。

【0128】

本実施の形態においては、PMMが各項目の情報ブロックを保持するように構成されていたが、PMMが表形式データをそのまま保持するような場合でも、上記G0rdは、後述するように同様に機能する。

【0129】

たとえば、本実施の形態においてコンパイルが終了した状態（たとえば、図17参照）で、グローバル順序集合配列G0rdの値の順序で、各項目の項目値を取り出していくことにより、表形式データ全体のビューを作成することができる。検索が終了した状態（たとえば、図26参照）やソートが終了した状態（たとえば、図56参照）においても、同様である。

【0130】

より詳細には、たとえば、図17において、PMM-2の制御回路20が、順位を示す値「5」を受理すると、グローバル順序集合配列G0rd中の値「5」に関連する（ローカルな）順序集合配列OrdSet中の値「0」が特定される。さらに、項目「年齢」に関して、ポインタ配列PV中の値「1」が特定され、次いで、値リストVL中の項目値「20」を特定することができる。無論、他の項目についても、ポインタ配列PV中の値、および、当該配列PV中の値にて特定される値リストVL中の項目値が特定される。これにより、順位を示す値に対応するレコードを取り

出すことが可能となる。

【0131】

また、情報ブロックを保持しないような構成であっても、上述したような、順位を示す値の受理に応答して、対応するレコードの取り出しを実現できる。これについては、図60を参照しつつ、後述する。

次に、情報ブロックを保持しないような構成を参照して、配列GOrdの意義について、さらに説明する。たとえば、図58（a）に示すように、表形式データを、値（項目値）そのものをソートするのではなく、項目を特定するアドレス情報となる順序集合配列の値をソートして、配列中の値を再配置することにより実現する場合を考える。図58（b）における順序集合配列OrdSetが、ソート後のレコードの順序を示している。

【0132】

次に、上記順序集合配列OrdSetおよび表形式データの本体（図58（b）の符号5800参照）を、複数のPMMにて分掌把握することを考える。順序集合配列OrdSetを分割し、かつ、表形式データ本体を分割して、分割された配列OrdSetおよび表形式データ本体の組を、PMMに分掌させた例を図59に示す。この場合、あるPMM配列OrdSet中の値が、他のPMMが保持するレコードを指示する場合もある（たとえば、矢印5901、5902参照）。したがって、各PMMにて単独で実行できる処理が実質上存在しない。また、図59の例で、さらに、項目「性別」を「女性」という項目値に絞り込む（検索する）場合に、絞り込まれたレコードを示す値を収容する配列を、どのように分掌すれば良いか、明確な基準を作ることができない。つまり、上述した状態での検索は実質的に不可能となる。

【0133】

これに対して、図60に示すように、順序集合配列OrdSetによって、各PMMが把握する表形式データの部分集合における、ローカルなソートされたレコードの順位を掌握し、かつ、グローバル順序集合配列GOrdが、ソートされたレコードのそれぞれの、全体における順位を掌握している。（ローカルな）順序集合配列OrdSetは、自己の掌握する表形式データの部分集合のレコードを指示するため、

P MM単独での処理が可能となる。

【0134】

図60に示す例における、順位を示す値の受理に応答したレコードの取り出しについても、以下に説明する。たとえば、P MM-0の制御回路が、順位を示す値「5」を受理すると、グローバル順序集合配列G0rd中の値「5」に関連する（ローカルな）順序集合配列0rdSet中の値「1」が特定される。これにより、P MM-0内の、「性別：男」、「年齢：21」、「身長：172」および「体重：64」というレコードが取り出される。

【0135】

さらに、特に、ここで注目すべきは、（ローカルな）順序集合配列0rdSetの値は、ローカルなソートが反映されるため、値の順位の逆転が生じ得るのに対して、グローバル順序集合配列G0rdの値が昇順になっていることである。これにより、高速なP MM間の処理およびP MM内の処理が可能となる。無論、検索処理やクロス集計処理の後においても、グローバル順序集合配列G0rdの値は昇順になっている。

【0136】

このように配列G0rdが昇順であることは、以下のようない点を生じる。たとえば、先に説明した検索処理において、配列G0rd（処理に使用されるものは配列G0rd'）が昇順であるため、値の比較を高速に実現できる（図23～図25参照）。同様に、先に説明したソート処理においても、配列G0rd（処理に使用されるものは配列G0rd'）が昇順であるため（これに加えて、後述する配列GVNoも昇順であるため）、値の比較処理を高速に実現できる（図52～図54参照）。

【0137】

また、ソートされたレコードを取り出して、ソートされたビューを作成する場合にも、各P MMは、グローバル順序集合配列G0rdが昇順であるから、先頭のレコードから順に、データを出力していけばよいため、処理を高速化することができる。

【0138】

次に、グローバル値番号配列GVNoの意義について説明する。グローバル値番号

配列は、集計の際に特に有用である。たとえば、各PMMにおいて、図61に示すような表形式データの部分集合のレコードが掌握されていると考える（符号6100～6103参照）。このような例において、各PMMでの集計結果を統合して、表形式データ全体の集計結果を取得することを考える。

【0139】

たとえば、項目「年齢」の出現数を集計する場合、各PMMでは、項目「年齢」の項目値ごとに、その出現数を示すような配列を作ることができる（符号6110～6113参照）。しかしながら、各PMMにおける各項目値の出現数をそのまま統合することはきわめて困難である。

これに対して、グローバル値番号配列を導入することにより、各PMMにおいて保持される項目値の、一定の順序の下での、項目値全体における位置を特定することができる。したがって、このグローバル値番号配列の値をキーとして、全体の集計、つまり、出現値の統合を実現することが可能となる。

【0140】

さらに、クロス集計においては、本実施の形態にかかる、項目ごとの情報ブロックに設けられたグローバル値番号配列GVNoを利用し、クロス集計にかかる項目のグローバル番号配列GVNoの値の組により一意的に特定される論理座標配列中の位置に、各PMMにおける集計結果を累算していくことで、出現値の統合を円滑に実現することが可能となる。

【0141】

また、グローバル値番号配列GVNoを利用して、項目値を特定して取り出すこと、つまり、グローバルな表形式データにおける値（項目値）の順位を示す情報を、PMMが受理し、その順位に対応する項目値を取り出すことも有用である。たとえば、図17において、項目「年齢」に関して、先頭、つまり、第0番の項目値を知るために、値の順位「0」を示す命令を受理したPMM-1が、グローバル値番号配列GVNo中の値「0」に関連する値リストVL中の値「16」を特定することができる。無論、命令を受理したPMM-2が同様に動作しても良い。さらに、グローバル値番号配列GVNoも昇順となる。これは、各PMMにて掌握される（ローカルな）値リストの項目値が昇順であれば、その順序は保存されるから

である。したがって、上記ソート処理において、値の比較処理を高速に実現できる（図52～図54参照）。

【0142】

本発明は、以上の実施の形態に限定されることなく、特許請求の範囲に記載された発明の範囲内で、種々の変更が可能であり、それらも本発明の範囲内に包含されるものであることは言うまでもない。

【0143】

前記実施の形態においては、PMMを、一方が時計回りにパケットを伝送する第1のバス（第1の伝送路）、他方が反時計回りにパケットを伝送する第2のバス（第2の伝送路）にて、リング状に接続している。このような構成により、パケット伝送の遅延時間などを均一化することができるため有利である。しかしながら、これに限定されず、バス型など他の形態の伝送路を採用しても良い。

【0144】

また、本実施の形態においては、メモリ、インターフェースおよび制御回路を有するPMMを利用しているが、これに限定されるものではなく、パーソナルコンピュータ、サーバなどを、ローカルな表形式データを分掌する情報処理ユニットとして、PMMの代わりに利用しても良い。或いは、単一のパーソナルコンピュータやサーバが、複数の情報処理ユニットを保持するような構成を採用しても良い。これらの場合でも、情報処理ユニットが、レコードの順位を示す値を受理し、グローバル順序集合配列GOrdを参照することにより、レコードを特定することができる。また、グローバル値番号配列を参照することにより、項目値を特定することも可能である。

【0145】

また、情報処理ユニット間の伝送路も、いわゆるネットワーク型やバス型を採用しても良い。

【0146】

単一のパーソナルコンピュータに複数の情報処理ユニットを設けるような構成を採用することで、以下のように、本発明を利用することができる。たとえば、札幌支社、東京支社、福岡支社の3つの表形式データを用意し、通常は、各支社

の単位で、検索、集計、ソートなどを実行する。さらに、3つの支社を統合したグローバルな表形式データを考えて、各支社の表形式データが、全体表のうちの部分表であるとみなし、グローバルな表形式データに関する検索、ソートおよび集計を実現することができる。

【0147】

無論、複数のパーソナルコンピュータをネットワークにて接続した場合にも、同様に、パーソナルコンピュータにて分掌されるローカルな表形式データに関する処理、および、グローバルな表形式データに関する処理を実現することもできる。

【0148】

【発明の効果】

本発明によれば、分散メモリ型において、処理と通信を統合することで著しく高速な並列処理を実現可能な情報処理装置を提供することが可能となる。

【図面の簡単な説明】

【図1】 図1は、本発明の実施の形態にかかる情報処理システムの概略を示すブロックダイヤグラムである。

【図2】 図2は、本発明の実施の形態にかかるPMMの構造の一例を示す図である。

【図3】 図3は、表形式データの一例を示す図である。

【図4】 図4は、本実施の形態において、表形式データを保持する構造の原理を説明するための図である。

【図5】 図5は、本実施の形態において、各PMMにて分掌把握される配列およびその値を説明する図である。

【図6】 図6は、初期的にPMM-0～4の各々にてそれぞれ分掌される表形式データの例を示す図である。

【図7】 図7は、初期的にPMM-0～4の各々にてそれぞれ分掌される表形式データの例を示す図である。

【図8】 図8は、本実施の形態にかかるコンパイル処理を概略的に示すフローチャートである

【図9】 図9は、図6～図7に示す例でのグローバル順序集合配列G0rdへの値の配置を示す図である。

【図10】 図10は、本実施の形態にかかるコンパイル処理におけるパケット伝送の例を示す図である。

【図11】 図11は、本実施の形態にかかるコンパイル処理におけるパケット伝送の例を示す図である。

【図12】 図12は、本実施の形態にかかるコンパイル処理におけるパケット伝送の例を示す図である。

【図13】 図13は、本実施の形態にかかるコンパイル処理におけるパケット伝送の例を示す図である。

【図14】 図14 (a)、(b)は、それぞれ、本実施の形態にかかるコンパイル処理における、パケット送出および受理の際にPMMにて実行される処理を示すフローチャートである。

【図15】 図15 (a)、(b)は、それぞれ、本実施の形態にかかるコンパイル処理における、パケット受理の際にPMMにて実行される処理を示すフローチャートである。

【図16】 図16は、本実施の形態にかかる検索処理の部分を概略的に示すフローチャートである。

【図17】 図17は、各PMMにおいて、値がセットアップされたフラグ配列および領域として新たなグローバル順序集合配列および順序集合配列が生成された状態の一例を示す図である。

【図18】 図18は、各PMMにおいて、図16の処理が実行され、ローカルにかつ並列的に、新たなグローバル順序集合配列G0rd' および順序集合配列OrdSet' に値が配置される状態の例を示す図である。

【図19】 図19は、配列中、不要な領域を削除した状態を示す図である。

【図20】 図20は、本実施の形態にかかる検索処理において、パケット伝送に先立って実行される処理を示すフローチャートである。

【図21】 図21は、本実施の形態にかかる検索処理におけるパケット伝送の例を示す図である。

【図22】 図22（a）、（b）は、それぞれ、本実施の形態にかかる検索処理における、パケット送出および受理の際にPMMにて実行される処理を示すフローチャートである。

【図23】 図23は、本実施の形態にかかる検索処理におけるパケット伝送の例を示す図である。

【図24】 図24は、本実施の形態にかかる検索処理における、パケット受理の際にPMMにて実行される処理を示すフローチャートである。

【図25】 図25は、本実施の形態にかかる検索処理におけるパケット伝送の例を示す図である。

【図26】 図26は、本実施の形態にかかる検索処理の結果得られた新たな配列である。

【図27】 図27は、本実施の形態にかかるクロス集計処理の部分を概略的に示すフローチャートである。

【図28】 図28は、クロス集計すべき項目について、各PMMにおいて、項目の値リストVLのサイズを乗じたサイズの領域が作られ、それに初期値「0」が与えられた状態を示す図である。

【図29】 図29は、クロス集計処理における、各PMMのカウントアップの一例を示す図である。

【図30】 図30は、それぞれのPMMにおいて、グローバル領域の値を特定するために、グローバル値番号配列GVNoの値を利用した状態を示す図である。

【図31】 図31（a）、（b）は、それぞれ、本実施の形態にかかるクロス集計処理における、パケット送出および受理の際にPMMにて実行される処理を示すフローチャートである。

【図32】 図32は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図33】 図33は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図34】 図34は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図35】 図35は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図36】 図36は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図37】 図37は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図38】 図38は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図39】 図39は、本実施の形態にかかるクロス集計処理におけるパケット伝送の例を示す図である。

【図40】 図40は、本実施の形態にかかるクロス集計処理による集計結果を示す図である。

【図41】 図41は、本実施の形態にかかるソート処理の部分を概略的に示すフローチャートである。

【図42】 図42は、ソートすべき項目について、各PMMにおいて、項目の値リストVLと同一のサイズを有する領域が作られ、それぞれに初期値「0」が与えられた状態を示す図である。

【図43】 図43は、各PMMにおけるカウントアップの一例を示す図である。

【図44】 図44は、本実施の形態にかかるソート処理の部分（累計数配列の生成）を概略的に示すフローチャートである。

【図45】 図45は、本実施の形態にかかる累計数配列の例を示す図である。

【図46】 図46は、本実施の形態にかかる、各PMMにて実行されるローカルなソート処理を示すフローチャートである。

【図47】 図47は、各PMMにおいてローカルなソート処理が実行されている状態の例を示す図である。

【図48】 図48は、各PMMにおいてローカルなソート処理が実行されている状態の例を示す図である。

【図49】 図49は、本実施の形態にかかるソート処理における、パケット送出の際にPMMにて実行される処理を示すフローチャートである。

【図50】 図50は、本実施の形態において、各PMMにおいて、初期値が配置された配列Cord'」が作られた状態を示す図である。

【図51】 図51は、本実施の形態にかかるソート処理におけるパケット伝送の例を示す図である。

【図52】 図52(a)、(b)は、それぞれ、本実施の形態にかかるソート処理における、パケット送出およびパケット受理の際にPMMにて実行される処理を示すフローチャートである。

【図53】 図53は、本実施の形態にかかるソート処理におけるパケット伝送の例を示す図である。

【図54】 図54は、本実施の形態にかかるソート処理におけるパケット伝送の例を示す図である。

【図55】 図55は、本実施の形態にかかるソート処理における、パケット受理の際にPMMにて実行される処理を示すフローチャートである。

【図56】 図56は、本実施の形態にかかるソート処理によるソート結果を示す図である。

【図57】 図57は、本実施の形態にかかるソート処理により得られた、項目「年齢」でソートされた表形式データの例を示す図である。

【図58】 図58は、表形式データのソートを、アドレス情報の再配置にて表現した例を示す図である。

【図59】 図59は、図58に示す表形式データを、グローバル順序集合配列無しに、各PMMにて分掌把握した例を示す図である。

【図60】 図60は、図58に示す表形式データを、グローバル順序集合配列を利用して、各PMMにて分掌把握した例を示す図である。

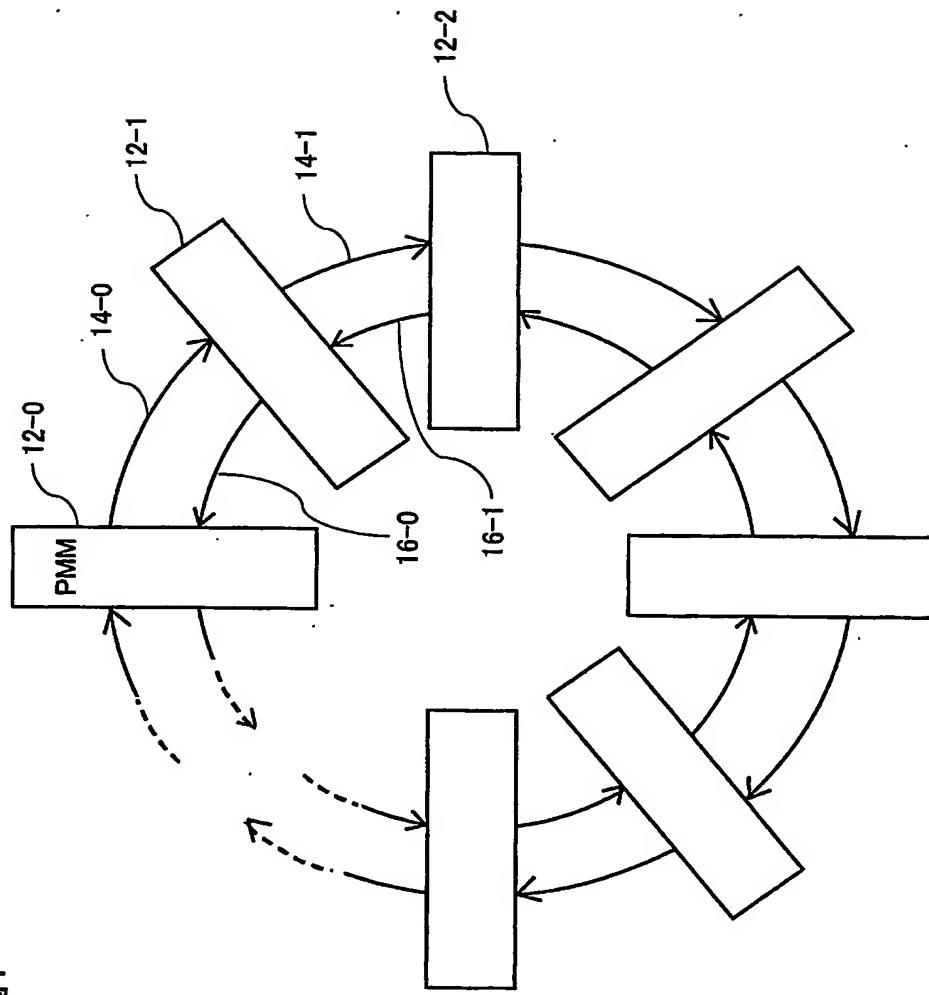
【図61】 図61は、各PMMにて集計結果を取得した状態を示す図である。

【符号の説明】

- 14 第1のバス
- 16 第2のバス
- 20 制御回路
- 22 バスI/F
- 24 メモリ
- 26 バンク

【書類名】 図面

【図1】



【図2】

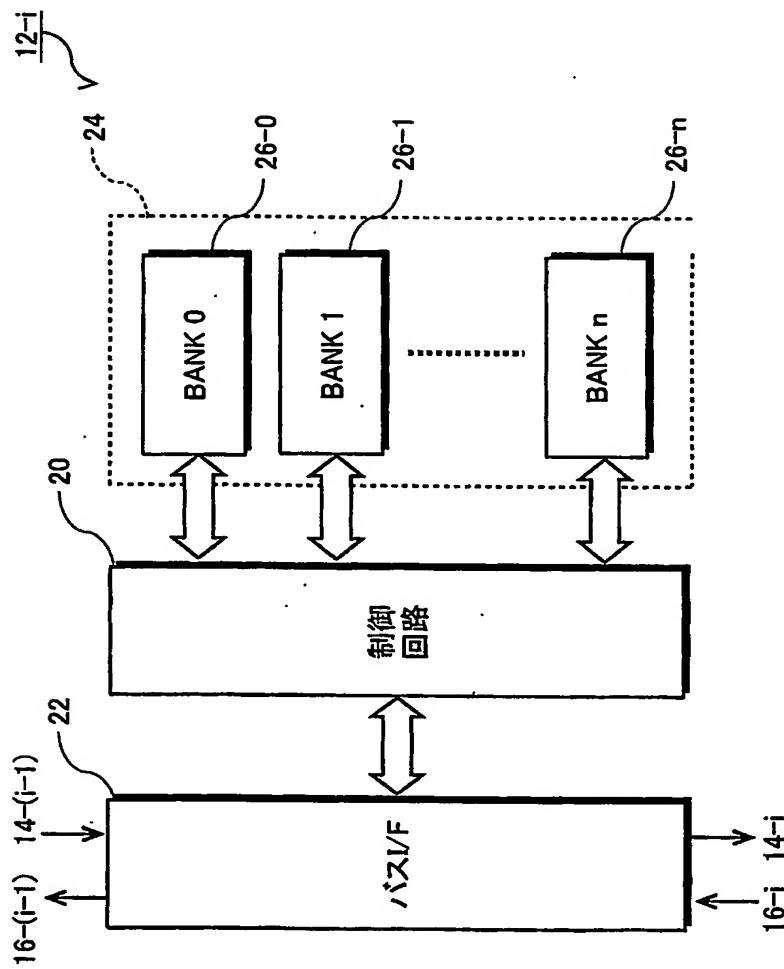


図2

【図3】

性別	年齢	身長(cm)	体重(kg)
女	18	168	55
男	21	172	64
女	24	159	48
女	16	172	48
男	28	181	78
女	20	166	55
女	16	168	52
男	33	174	65
男	24	177	64
女	18	170	55

図3

【図4】

性別		年齢		身長		体重	
VNo	VL	VNo	VL	VNo	VL	VNo	VL
0	男	0	16	0	159	0	48
1	女	1	18	1	166	1	52
2		2	20	2	168	2	55
3		3	21	3	170	3	64
4		4	24	4	172	4	65
5		5	28	5	174	5	78
6		6	33	6	177	6	
7		7	5	7	181	7	
8		8	6	8	6	8	
9		9	3	9	2	9	

昇順

OrdSet
0
1
2
3
4
5
6
7
8
9

図4

【図5】

図5

OFFSET= 0

PMM-0

	性別	年齢	身長(cm)	体重(kg)
0	女	18	168	55
1	男	21	172	64
2	女	24	159	48

OFFSET= 3

PMM-1

	性別	年齢	身長(cm)	体重(kg)
0	女	16	172	48
1	男	28	181	78

OFFSET= 5

PMM-2

	性別	年齢	身長(cm)	体重(kg)
0	女	20	166	55
1	女	16	168	52
2	男	33	174	65

OFFSET= 8

PMM-3

	性別	年齢	身長(cm)	体重(kg)
0	男	24	177	64
1	女	18	170	55

【図6】

性別(PMM-0)		年齢(PMM-0)		身長(PMM-0)		体重(PMM-0)	
Gord	OrdSet	VNo	VL	VNo	VL	VNo	VL
0	-	0	男	0	18	0	48
1	-	1	女	1	21	1	55
2	-	2	昇順	2	24	2	64
						昇順	
性別(PMM-1)		年齢(PMM-1)		身長(PMM-1)		体重(PMM-1)	
Gord	OrdSet	VNo	VL	VNo	VL	VNo	VL
0	-	0	男	0	16	0	48
1	-	1	女	1	28	1	78
						昇順	

図6

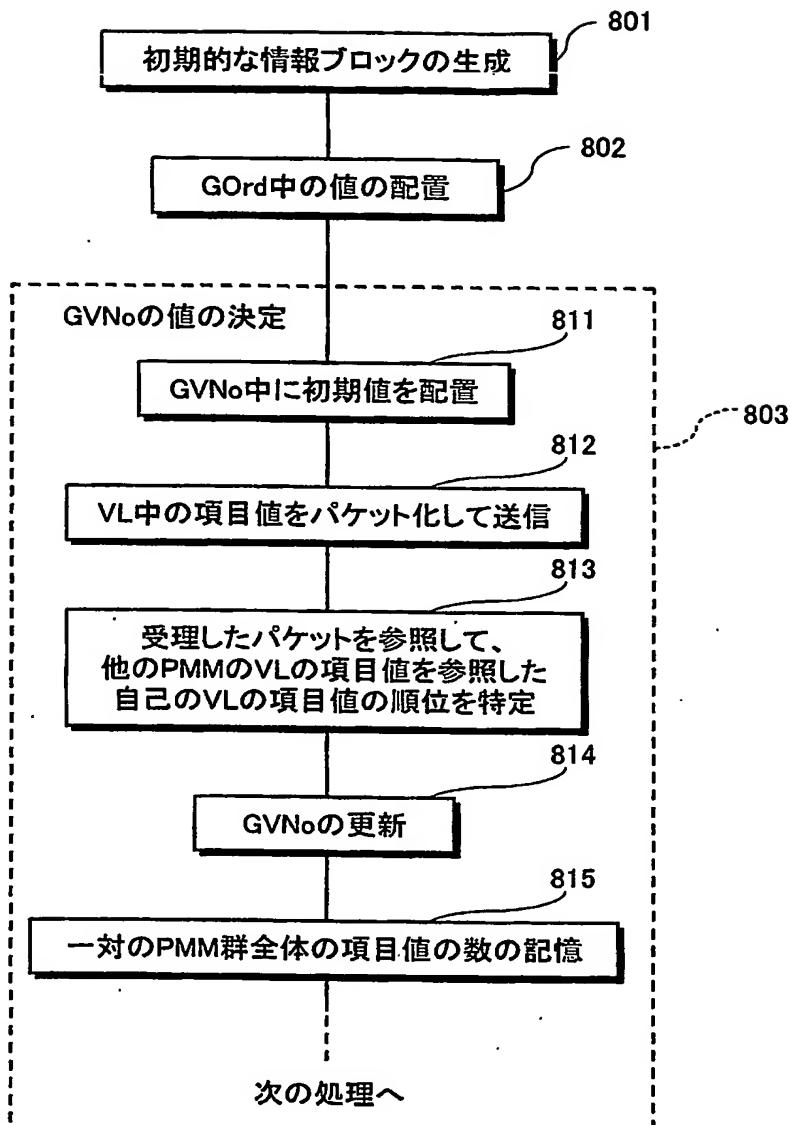
【図7】

性別(PMM-2)		年齢(PMM-2)		身長(PMM-2)		体重(PMM-2)	
Gord	OrdSet	VNo	VL	VNo	VL	VNo	VL
0	-	0	男	0	16	0	52
1	-	1	女	1	20	1	55
2	-	2	昇順	2	33	2	65
						昇順	
性別(PMM-3)		年齢(PMM-3)		身長(PMM-3)		体重(PMM-3)	
Gord	OrdSet	VNo	VL	VNo	VL	VNo	VL
0	-	0	男	0	18	0	55
1	-	1	女	1	24	1	64
						昇順	

図7

【図8】

図8



【図9】

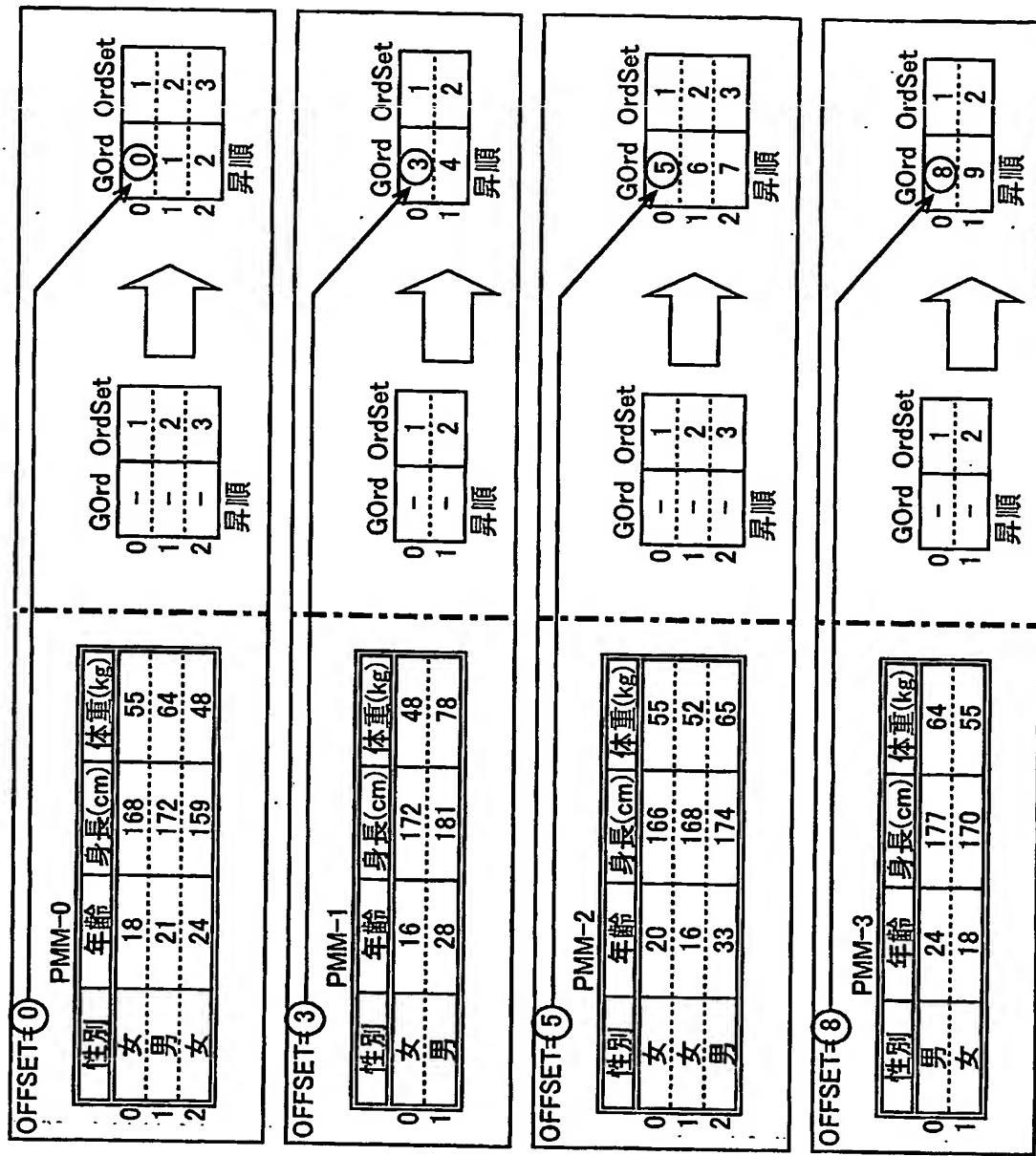


図9

【図10】

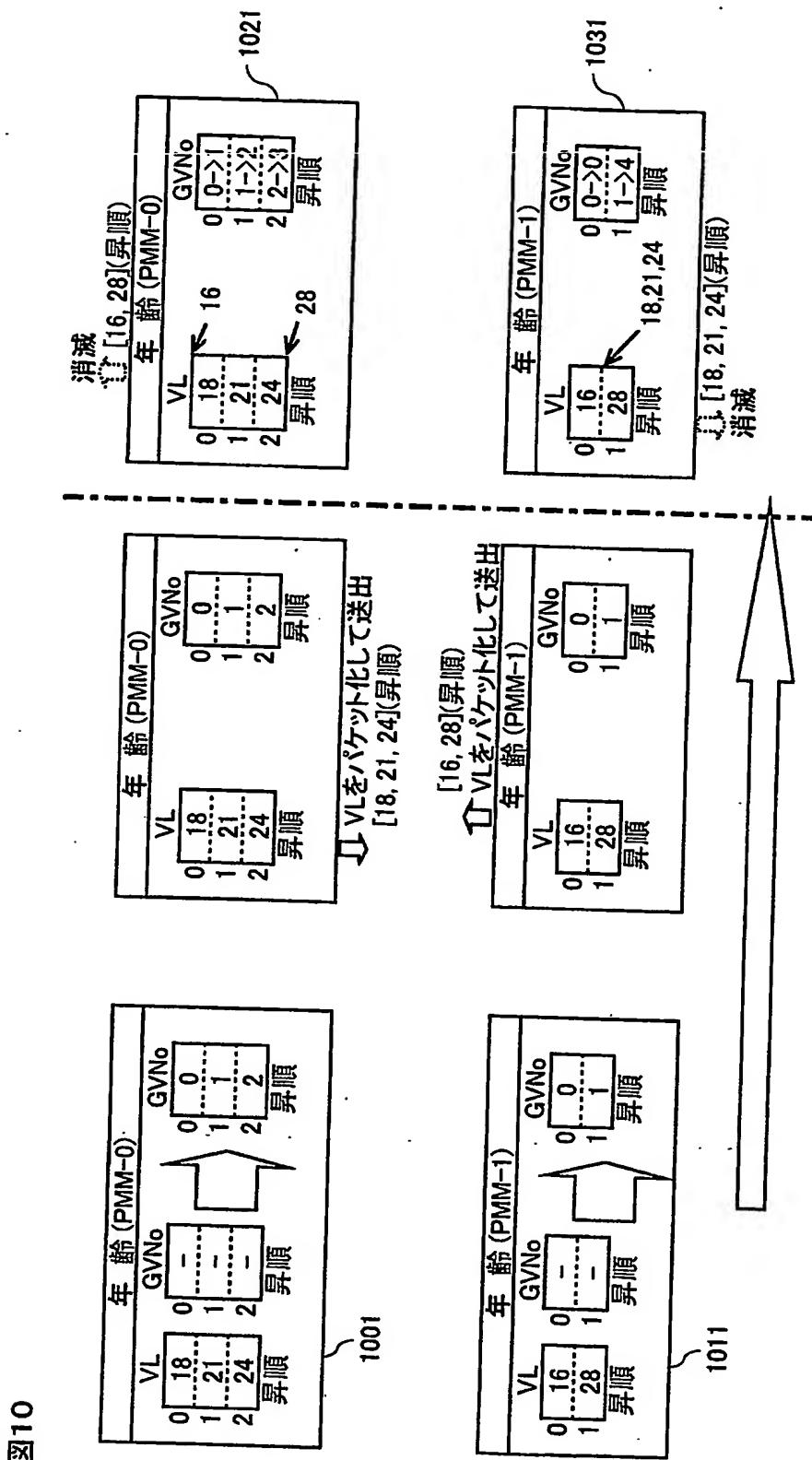
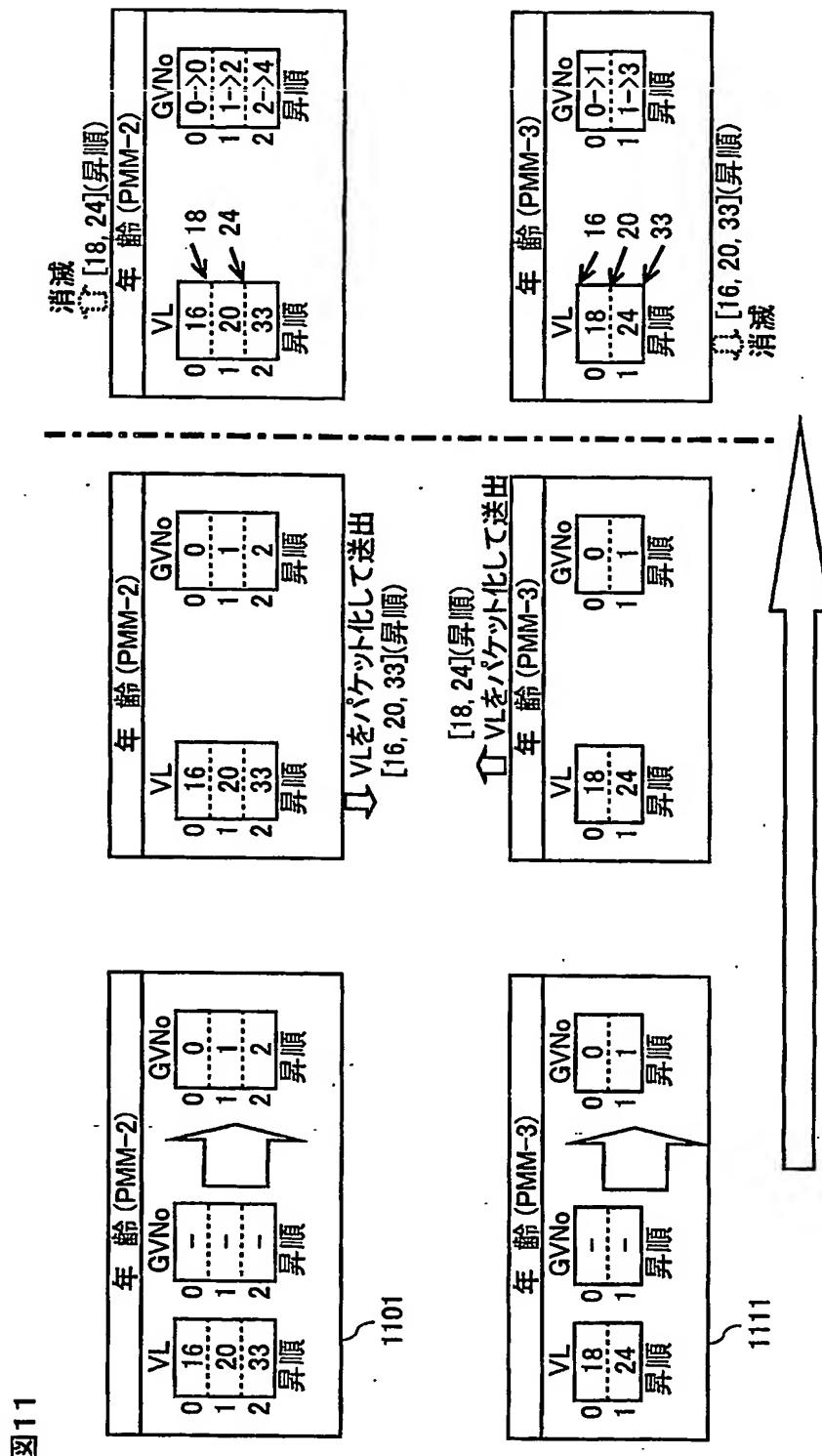


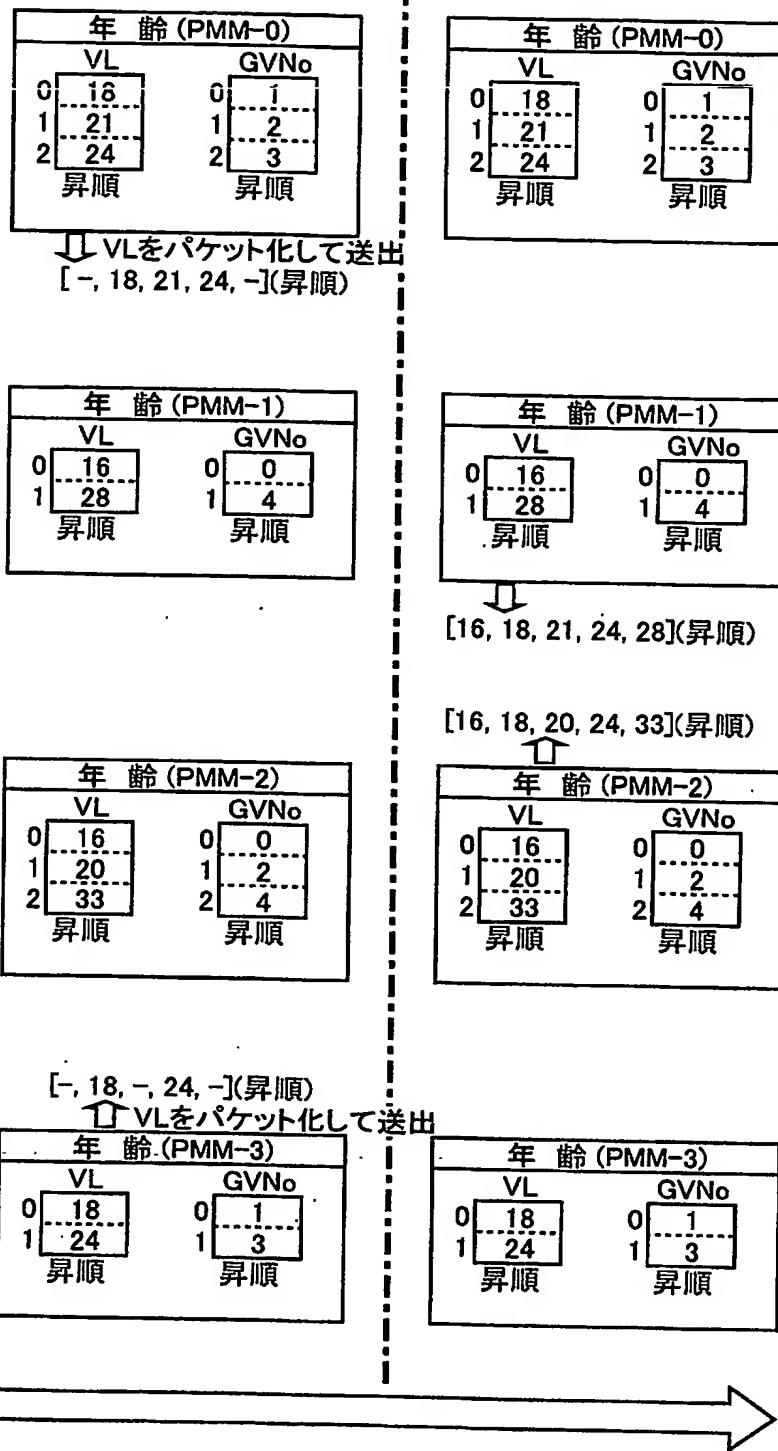
図10

【図11】



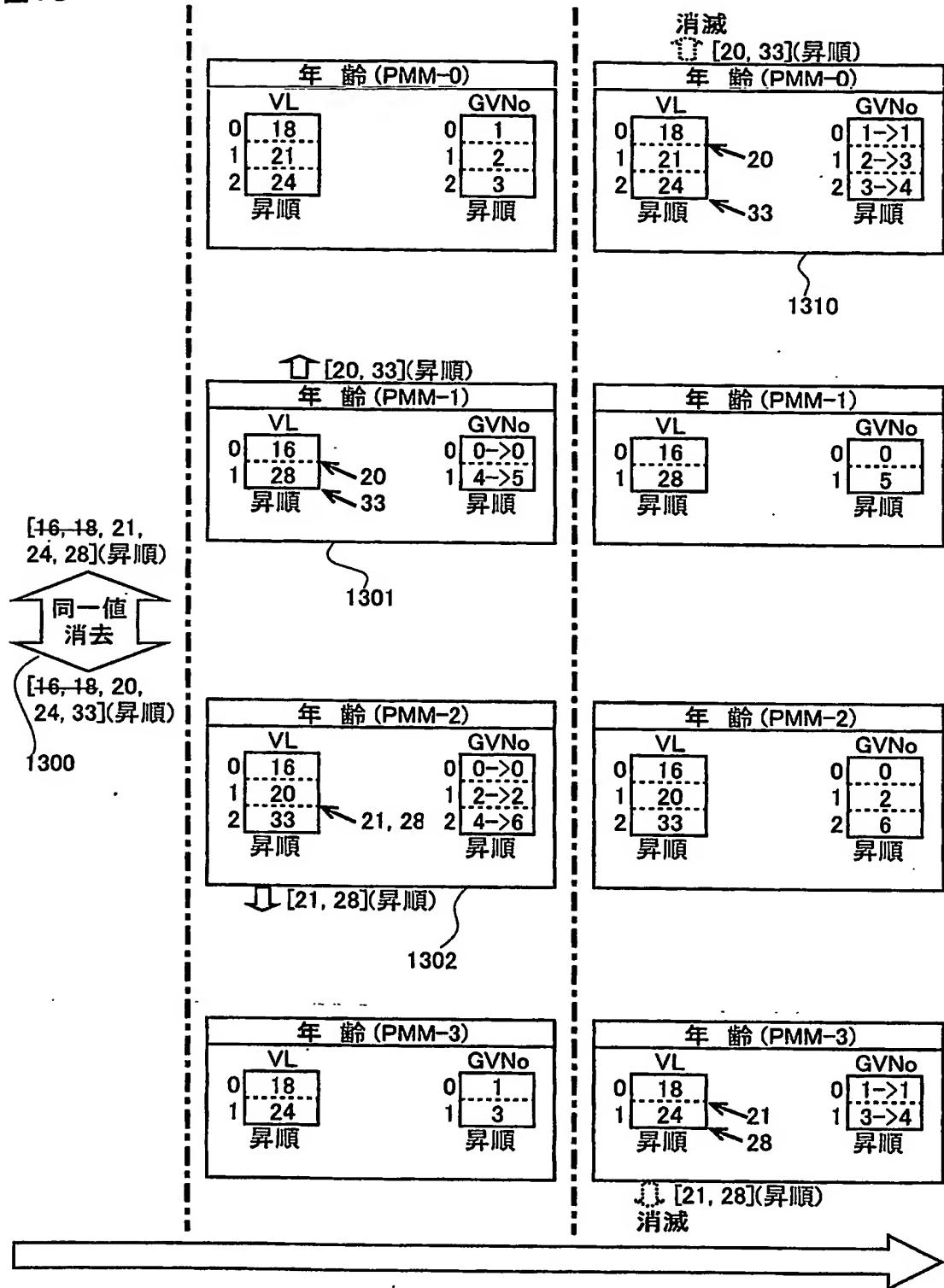
【図12】

図12



【図13】

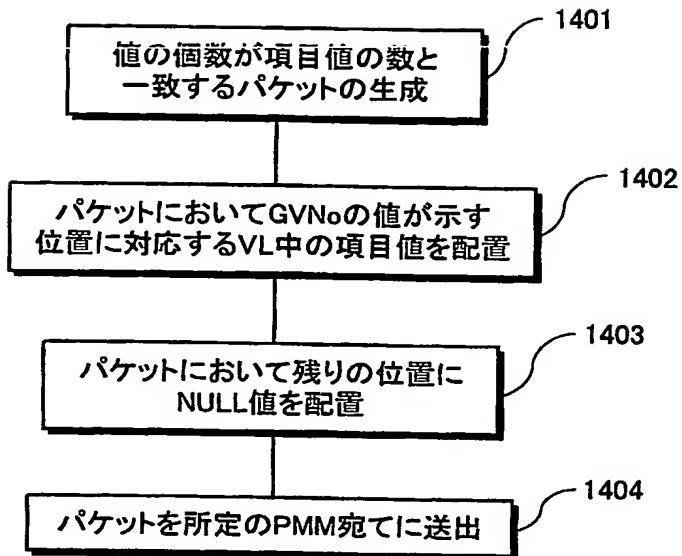
図13



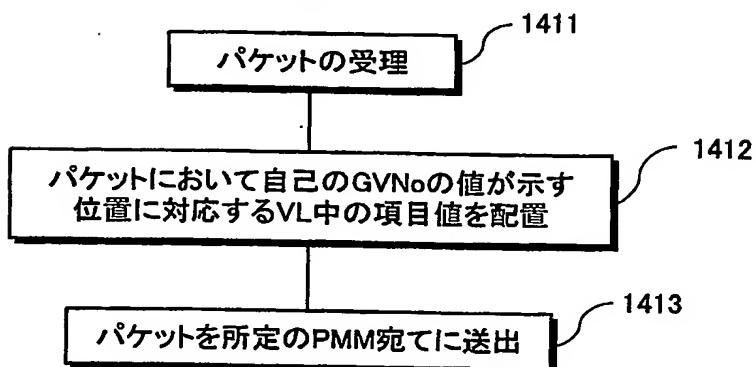
【図14】

図14

(a)

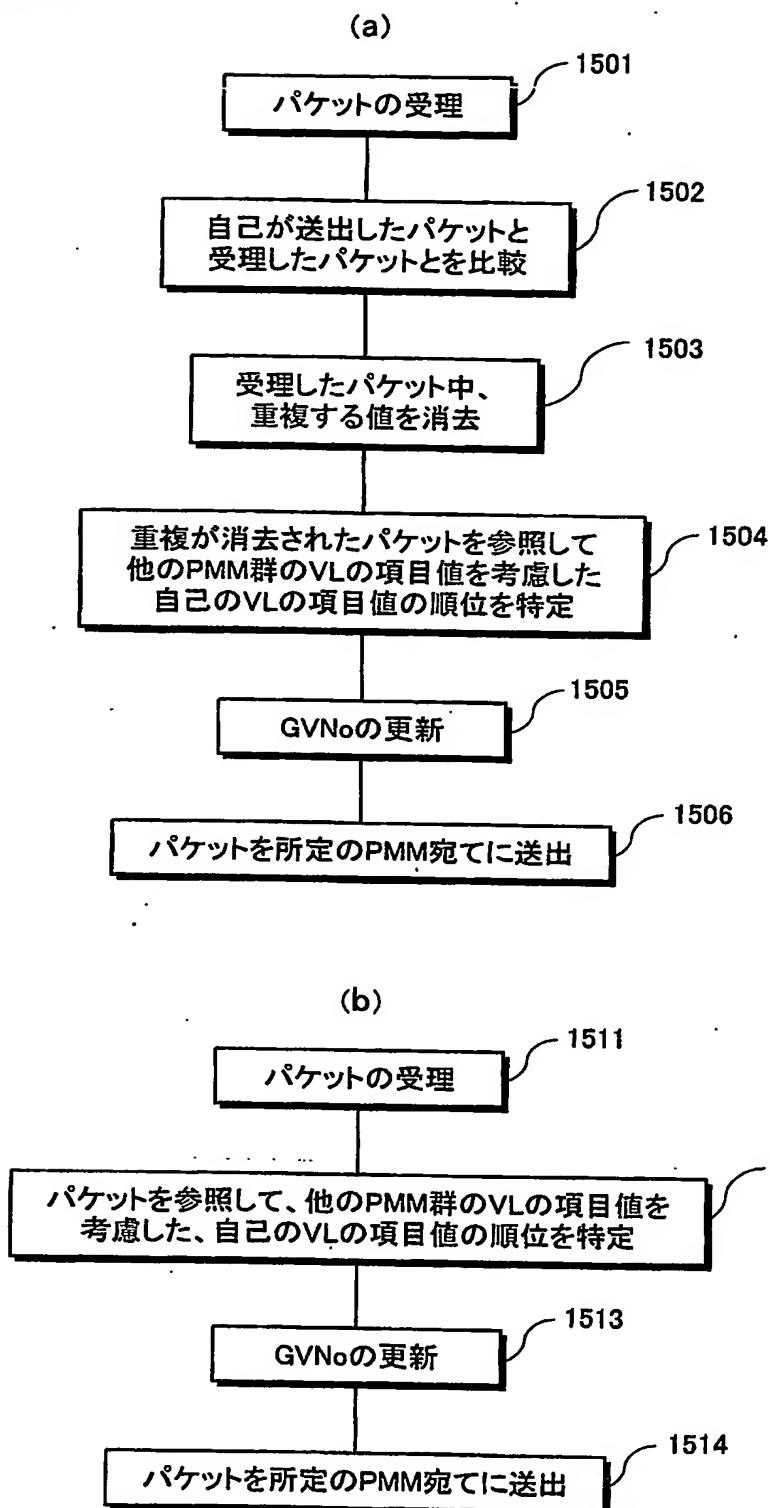


(b)



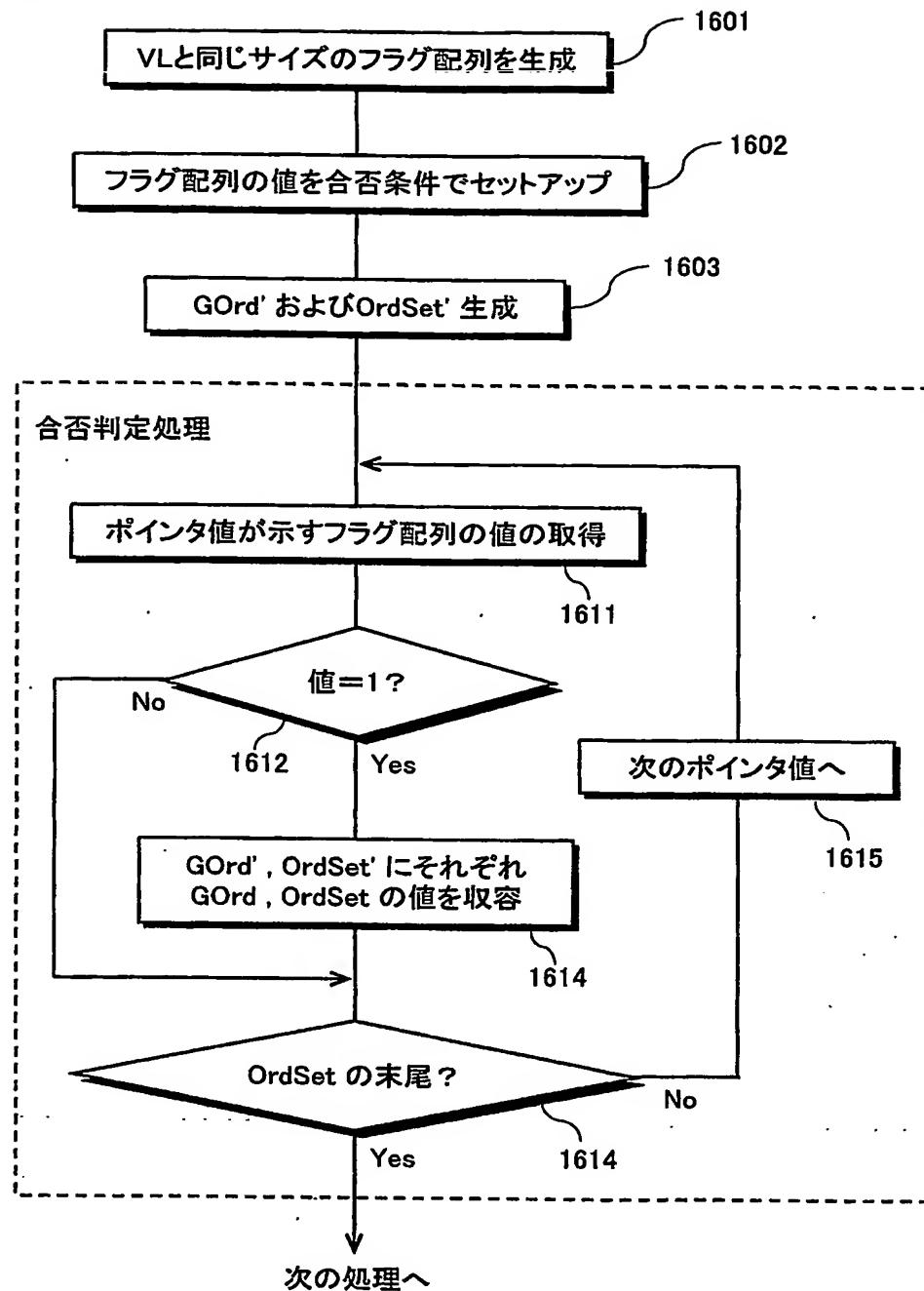
【図15】

図15



【図16】

図16



【図17】

図17

年齢(PMM-0)		GOrd' OrdSet'	
VNo	VL	フラグ配列	GVNo
0	0	0 0	0 1
1	1	1 1	1 3
2	2	2 2	2 4

昇順

年齢(PMM-1)		GOrd' OrdSet'	
VNo	VL	フラグ配列	GVNo
0	0	0 0	0 5
1	1	1 0	1 0

昇順

年齢(PMM-2)		GOrd' OrdSet'	
VNo	VL	フラグ配列	GVNo
0	0	0 0	0 0
1	0	1 20	1 2
2	2	2 33	2 6

昇順

年齢(PMM-3)		GOrd' OrdSet'	
VNo	VL	フラグ配列	GVNo
0	0	0 0	0 1
1	1	1 24	1 4

昇順

【図18】

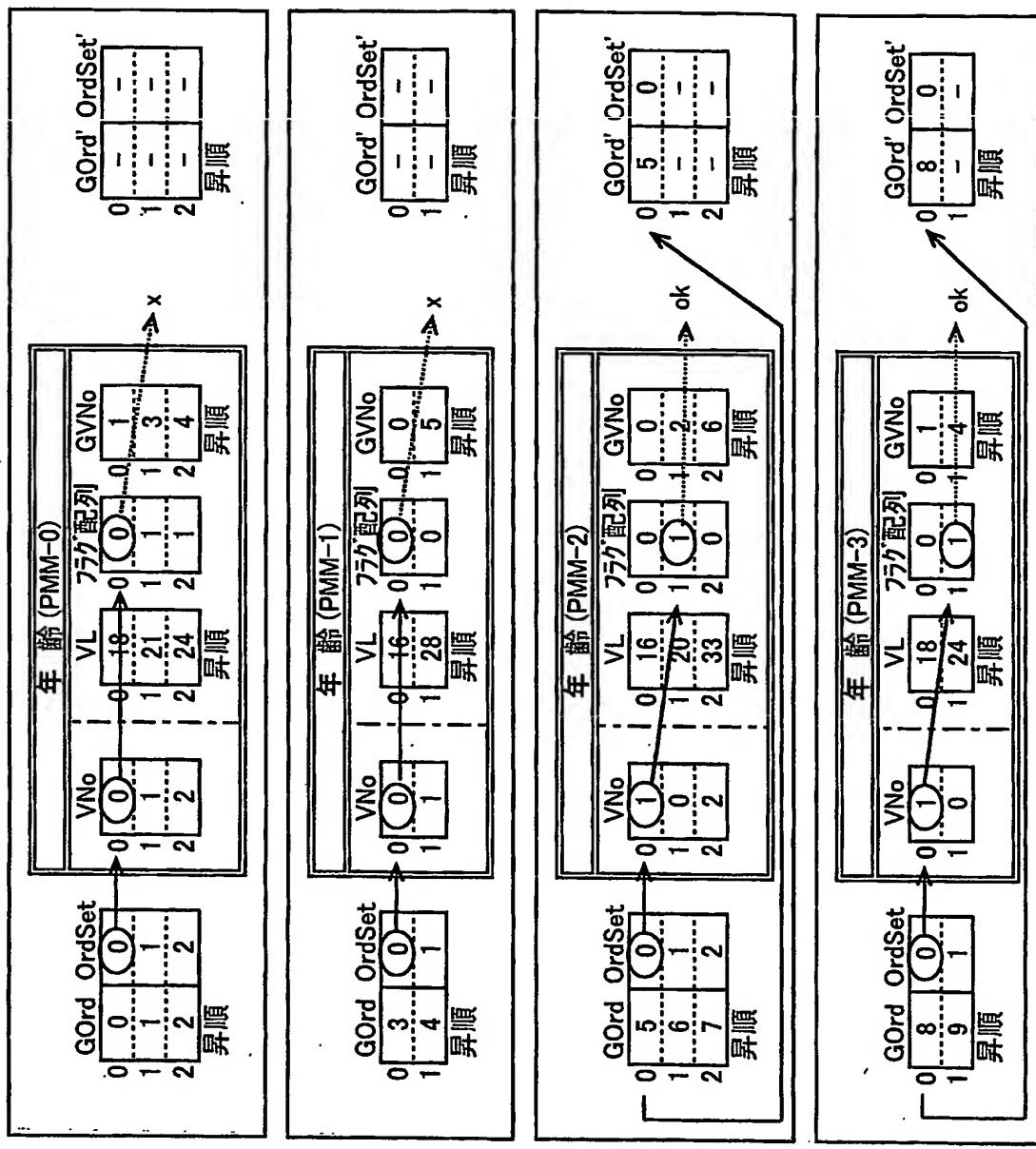
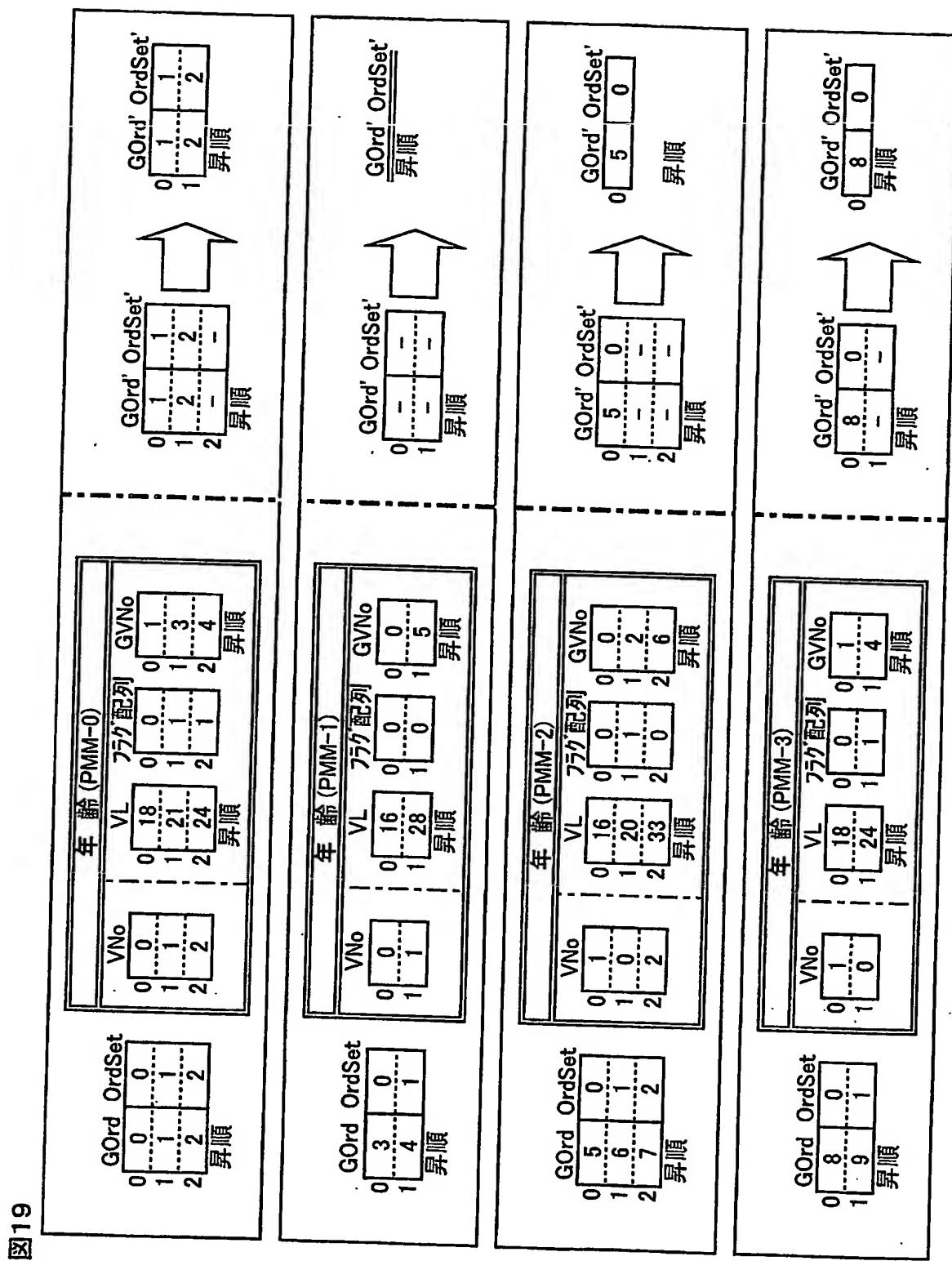


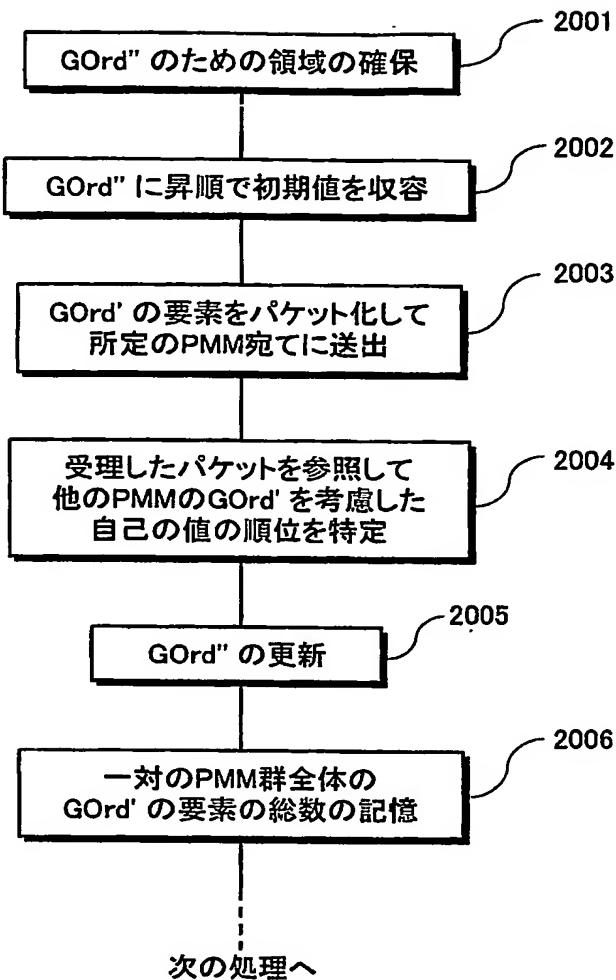
図18

【図19】



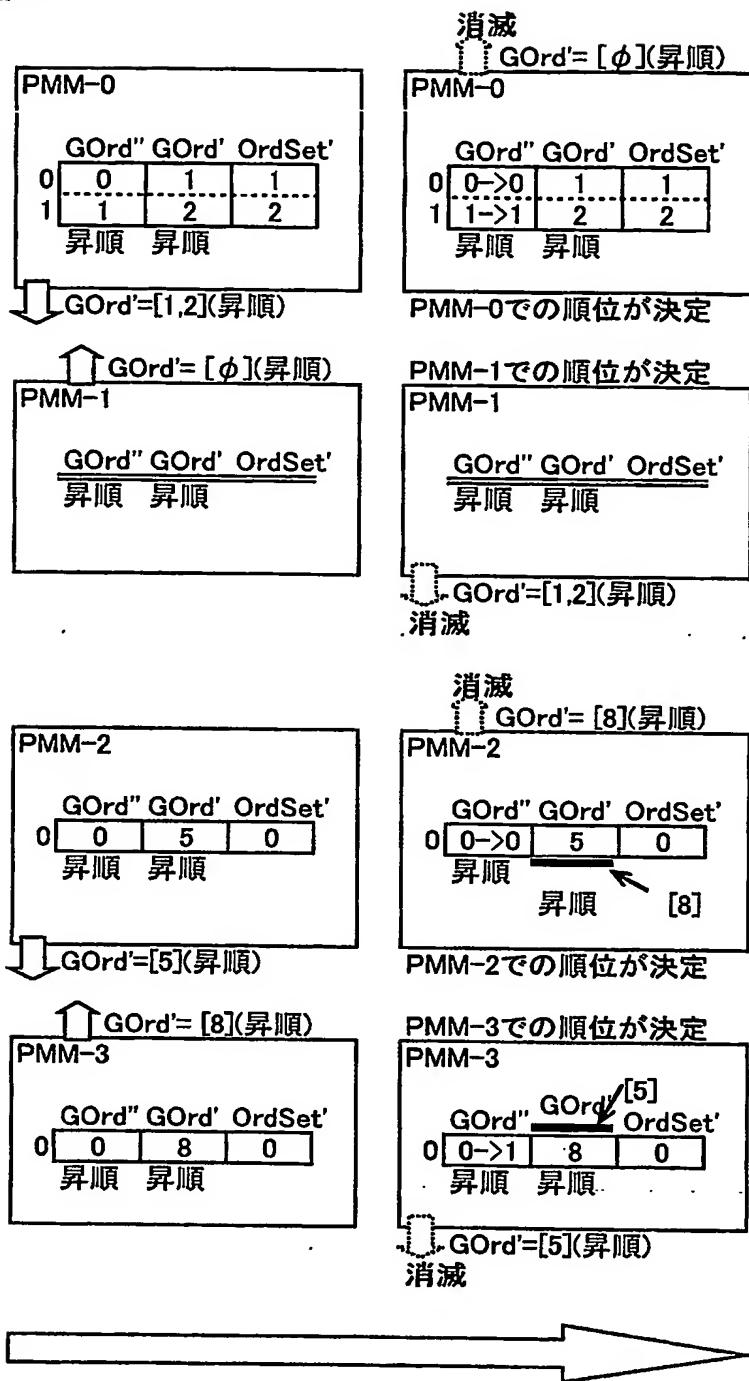
【図20】

図20



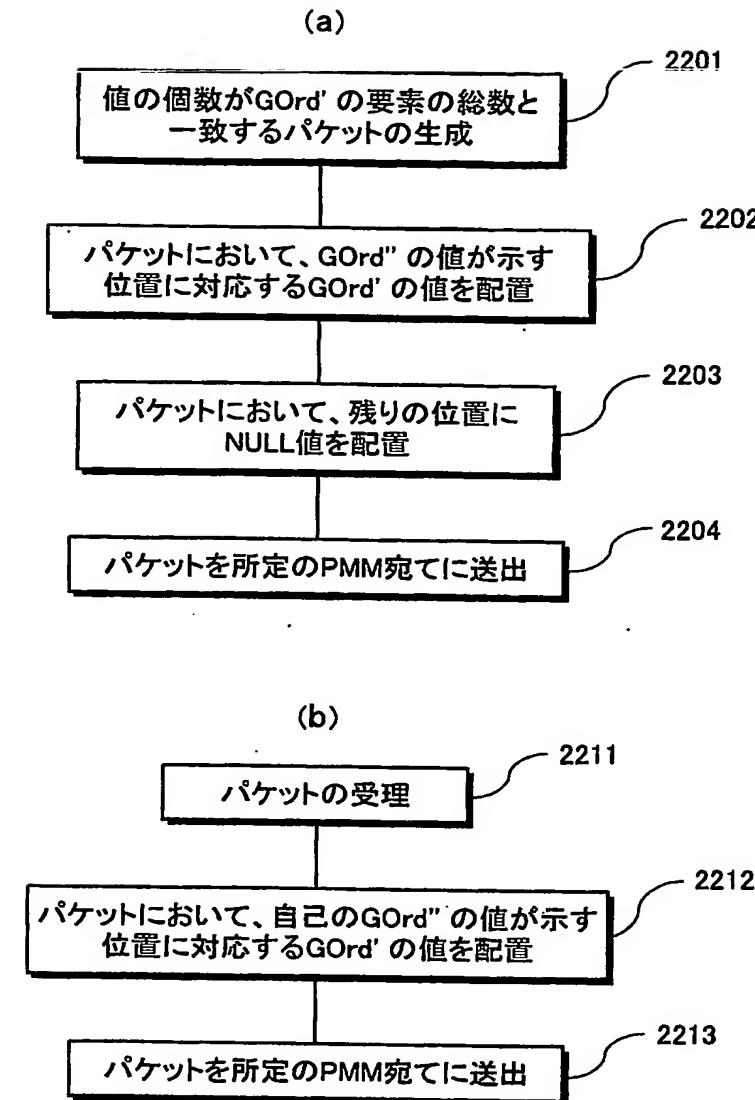
【図21】

図21



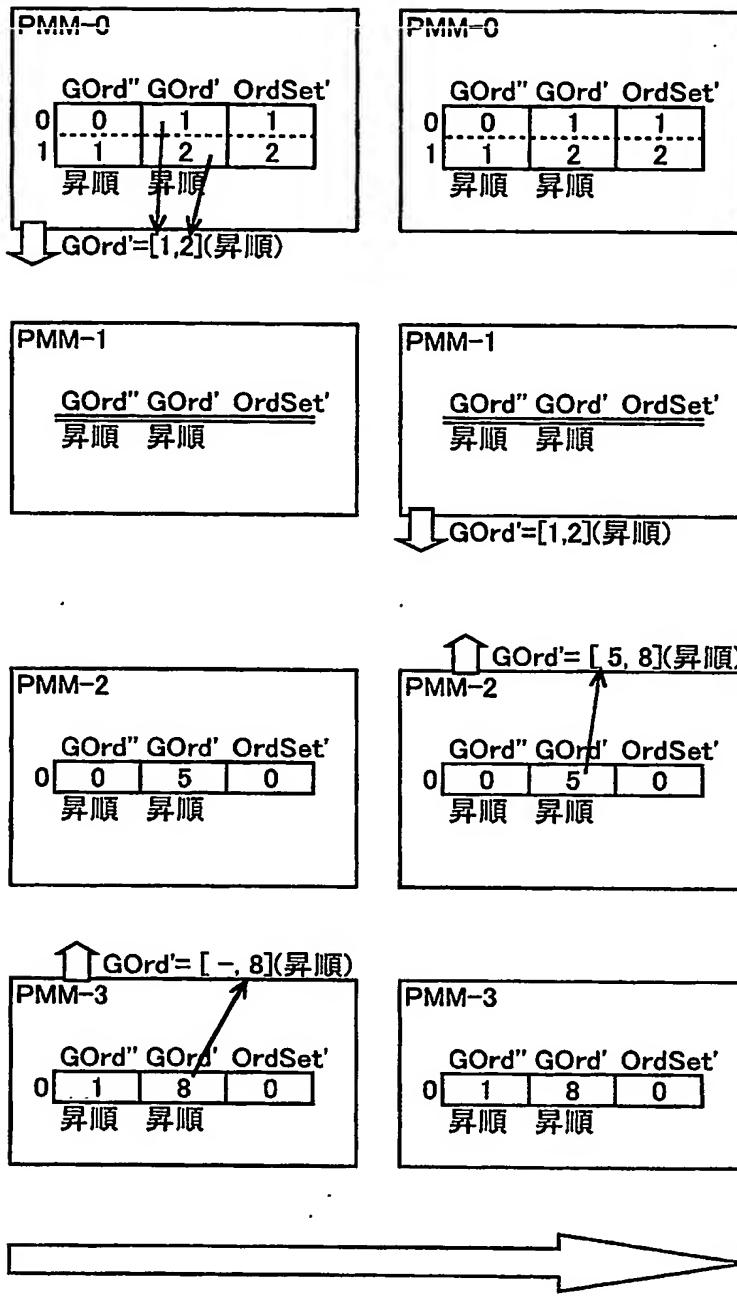
【図22】

図22



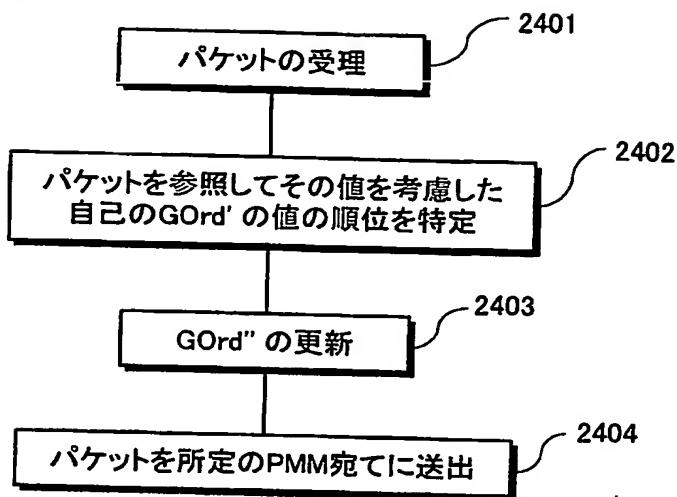
【図23】

図23



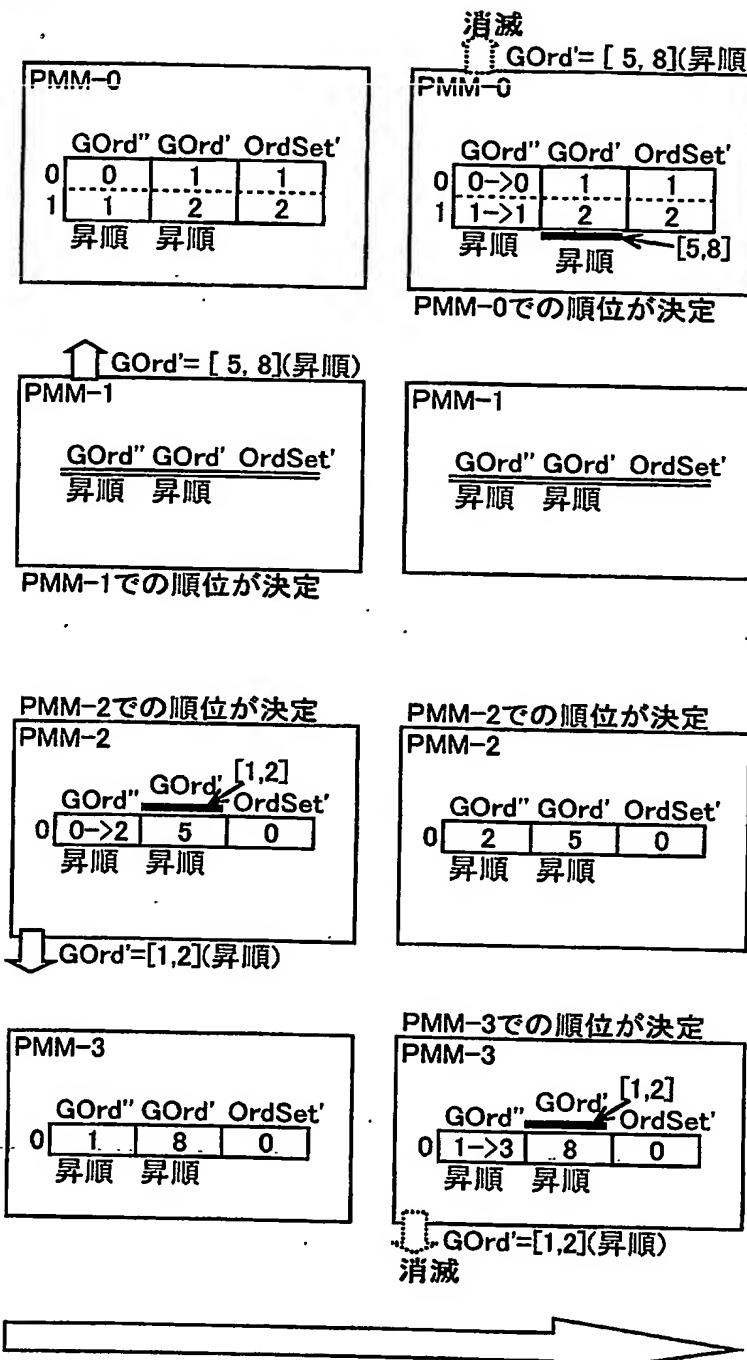
【図24】

図24



【図25】

図25



【図26】

図26

<u>GOrd</u> <u>OrdSet</u> 0 0 1 1 1 2 昇順	年齢 (PMM-0) <table border="1"> <thead> <tr> <th>VNo</th><th>VL</th><th>フラグ配列</th><th>GVNo</th></tr> </thead> <tbody> <tr> <td>0 0</td><td>0 18</td><td>0 0</td><td>0 1</td></tr> <tr> <td>1 1</td><td>1 21</td><td>1 1</td><td>1 3</td></tr> <tr> <td>2 2</td><td>2 24</td><td>2 1</td><td>2 4</td></tr> </tbody> </table> <p style="text-align: center;">昇順</p>				VNo	VL	フラグ配列	GVNo	0 0	0 18	0 0	0 1	1 1	1 21	1 1	1 3	2 2	2 24	2 1	2 4
VNo	VL	フラグ配列	GVNo																	
0 0	0 18	0 0	0 1																	
1 1	1 21	1 1	1 3																	
2 2	2 24	2 1	2 4																	
年齢 (PMM-1) <table border="1"> <thead> <tr> <th>VNo</th><th>VL</th><th>フラグ配列</th><th>GVNo</th></tr> </thead> <tbody> <tr> <td>0 0</td><td>0 16</td><td>0 0</td><td>0 0</td></tr> <tr> <td>1 1</td><td>1 28</td><td>1 0</td><td>1 5</td></tr> </tbody> </table> <p style="text-align: center;">昇順</p>				VNo	VL	フラグ配列	GVNo	0 0	0 16	0 0	0 0	1 1	1 28	1 0	1 5					
VNo	VL	フラグ配列	GVNo																	
0 0	0 16	0 0	0 0																	
1 1	1 28	1 0	1 5																	
<u>GOrd</u> <u>OrdSet</u> 0 2 0 昇順	年齢 (PMM-2) <table border="1"> <thead> <tr> <th>VNo</th><th>VL</th><th>フラグ配列</th><th>GVNo</th></tr> </thead> <tbody> <tr> <td>0 1</td><td>0 16</td><td>0 0</td><td>0 0</td></tr> <tr> <td>1 0</td><td>1 20</td><td>1 1</td><td>1 2</td></tr> <tr> <td>2 2</td><td>2 33</td><td>2 0</td><td>2 6</td></tr> </tbody> </table> <p style="text-align: center;">昇順</p>				VNo	VL	フラグ配列	GVNo	0 1	0 16	0 0	0 0	1 0	1 20	1 1	1 2	2 2	2 33	2 0	2 6
VNo	VL	フラグ配列	GVNo																	
0 1	0 16	0 0	0 0																	
1 0	1 20	1 1	1 2																	
2 2	2 33	2 0	2 6																	
年齢 (PMM-3) <table border="1"> <thead> <tr> <th>VNo</th><th>VL</th><th>フラグ配列</th><th>GVNo</th></tr> </thead> <tbody> <tr> <td>0 1</td><td>0 18</td><td>0 0</td><td>0 1</td></tr> <tr> <td>1 0</td><td>1 24</td><td>1 1</td><td>1 4</td></tr> </tbody> </table> <p style="text-align: center;">昇順</p>				VNo	VL	フラグ配列	GVNo	0 1	0 18	0 0	0 1	1 0	1 24	1 1	1 4					
VNo	VL	フラグ配列	GVNo																	
0 1	0 18	0 0	0 1																	
1 0	1 24	1 1	1 4																	

【図27】

図27

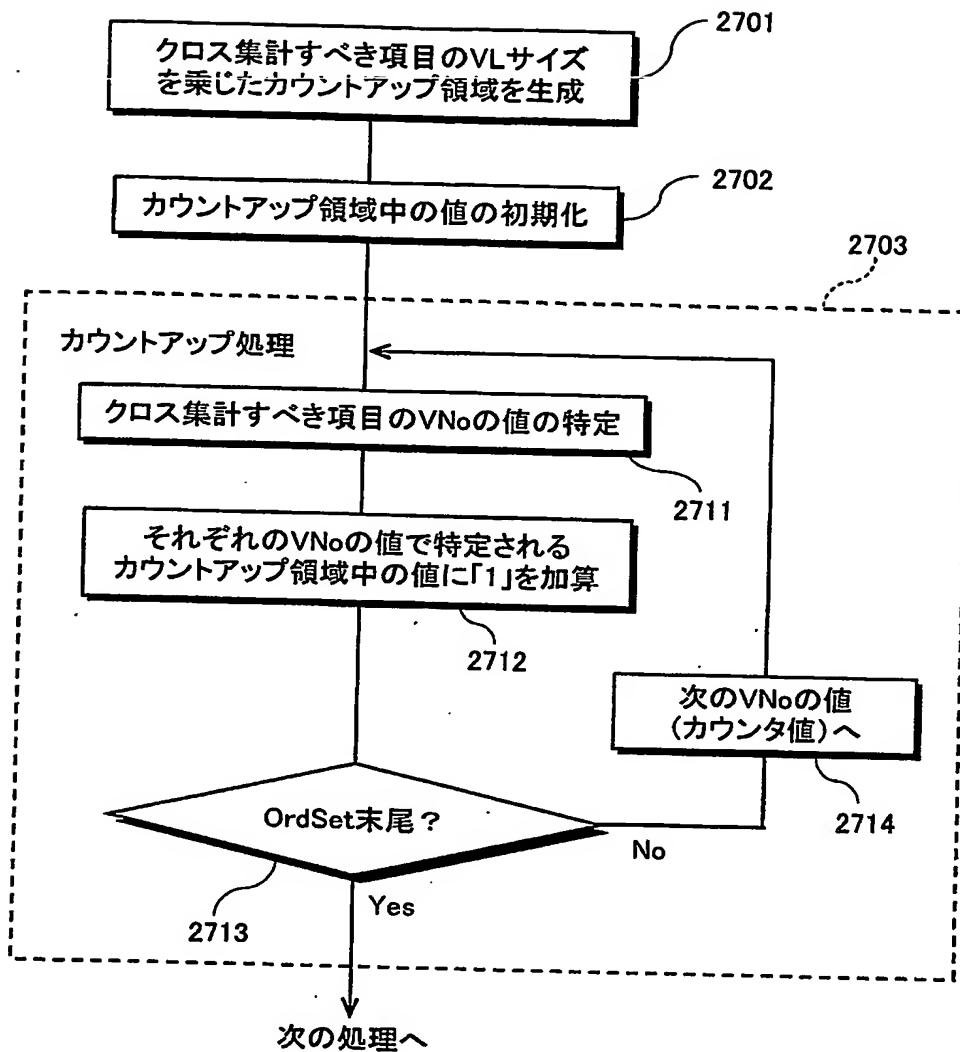


図28

【図28】

		性別(PMM-0)			年齢(PMM-0)				
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	年齢	
0	0	0	0	0	0	18	0	18	24
1	1	1	1	1	1	21	1	21	2
2	2	2	2	2	2	24	2	24	0
									昇順

		性別(PMM-1)			年齢(PMM-1)				
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	年齢	
0	3	0	0	0	0	16	0	16	28
1	4	1	1	1	1	28	1	28	0
									昇順

		性別(PMM-2)			年齢(PMM-2)				
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	年齢	
0	5	0	0	0	0	16	0	16	33
1	6	1	1	1	1	20	1	20	2
2	7	2	0	2	2	33	2	33	0
									昇順

		性別(PMM-3)			年齢(PMM-3)				
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	年齢	
0	8	0	0	0	0	18	0	18	24
1	9	1	1	1	1	24	1	24	0
									昇順

【図29】

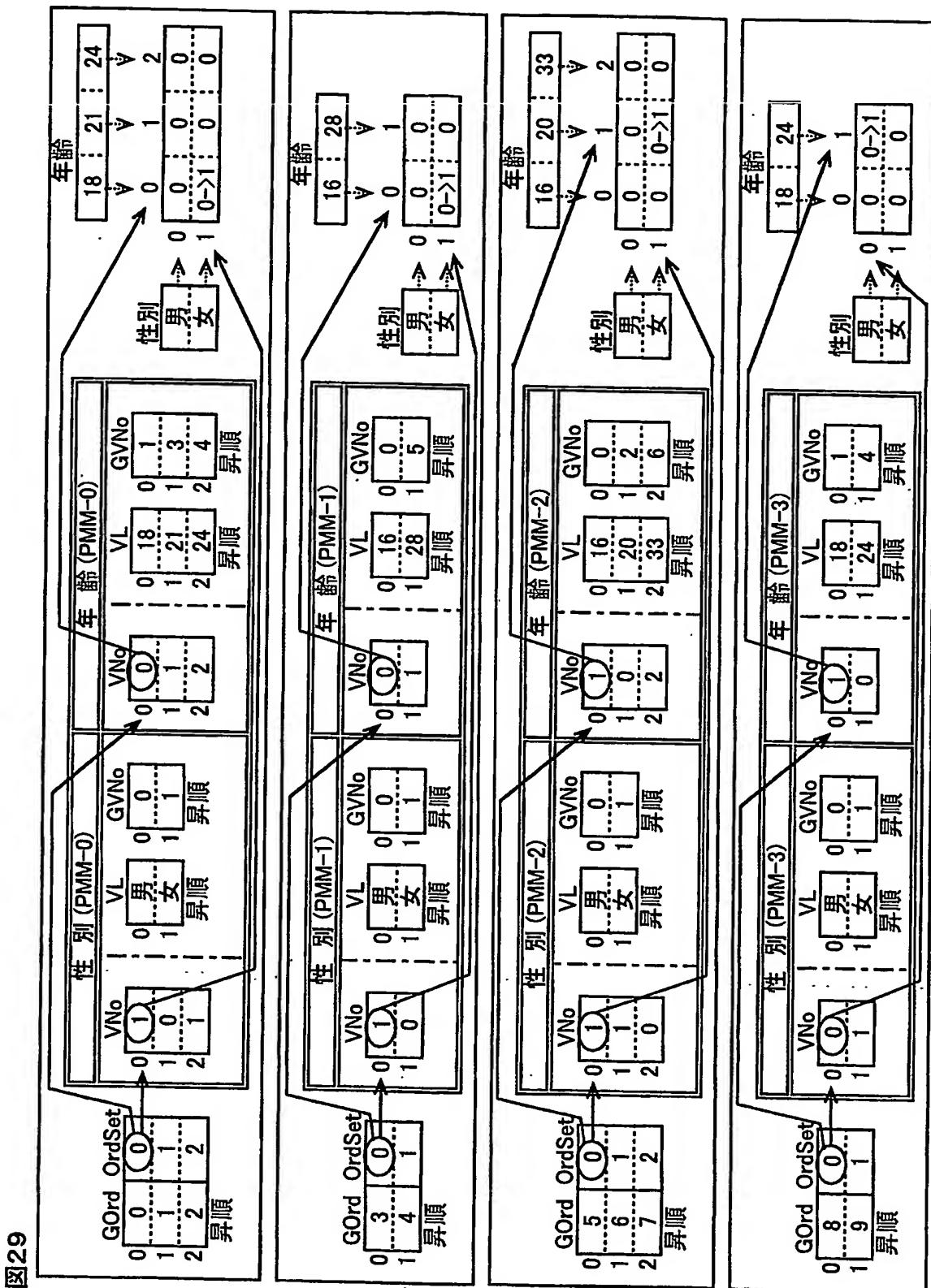


図29

【図 30】

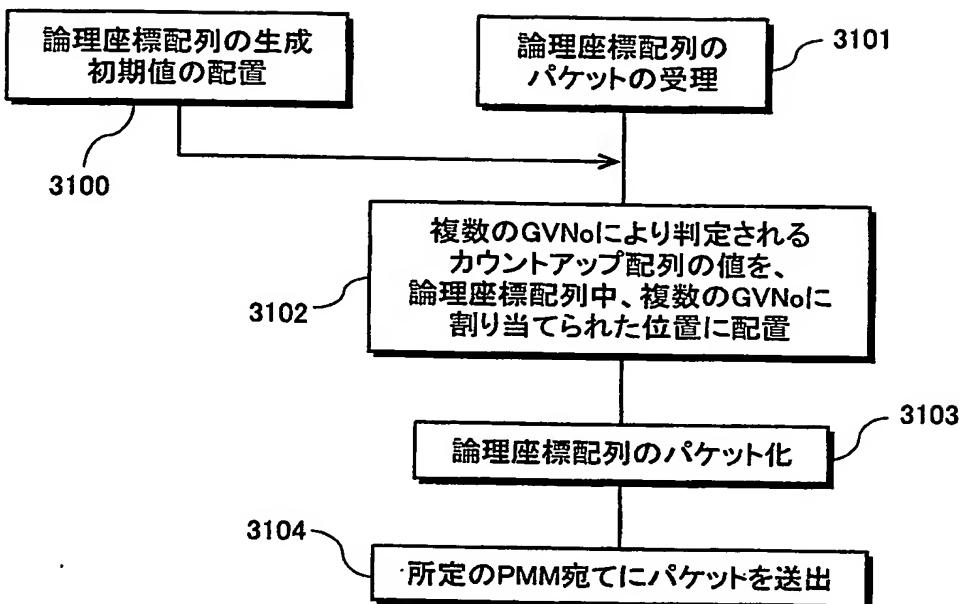
図 30

性別 (PMM-0)		年齢		性別 (PMM-1)		年齢		性別 (PMM-2)		年齢		性別 (PMM-3)	
VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL
0 1	男 0	0 1	1 3	0 1	1 18	0 1	1 16	0 1	1 16	0 1	1 18	0 1	1 18
1 0	女 1	1 0	0 1	1 0	2 1	1 0	2 28	1 28	1 20	1 0	1 24	1 0	1 24
2 1	昇順			2 1	2 4	2 1	2 33	2 33	2 33	2 0	1 4	1 0	1 4
性別 (PMM-0)		年齢		性別 (PMM-1)		年齢		性別 (PMM-2)		年齢		性別 (PMM-3)	
VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL	VNo	VL
0 1	男 0	0 1	1 5	0 1	1 16	0 1	1 16	0 1	1 16	0 1	1 18	0 1	1 18
1 0	女 1	1 0	0 1	1 0	2 28	1 28	2 33	2 33	2 33	2 0	1 4	1 0	1 4
2 1	昇順			2 1	2 0	2 1	2 0	2 1	2 0	2 0	1 0	1 1	1 0

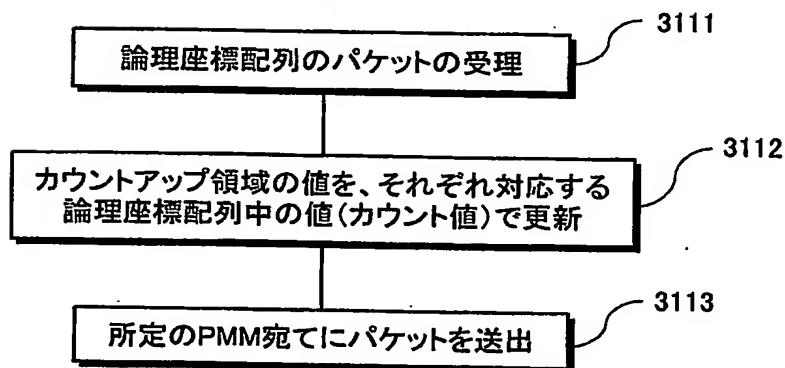
【図31】

図31

(a)

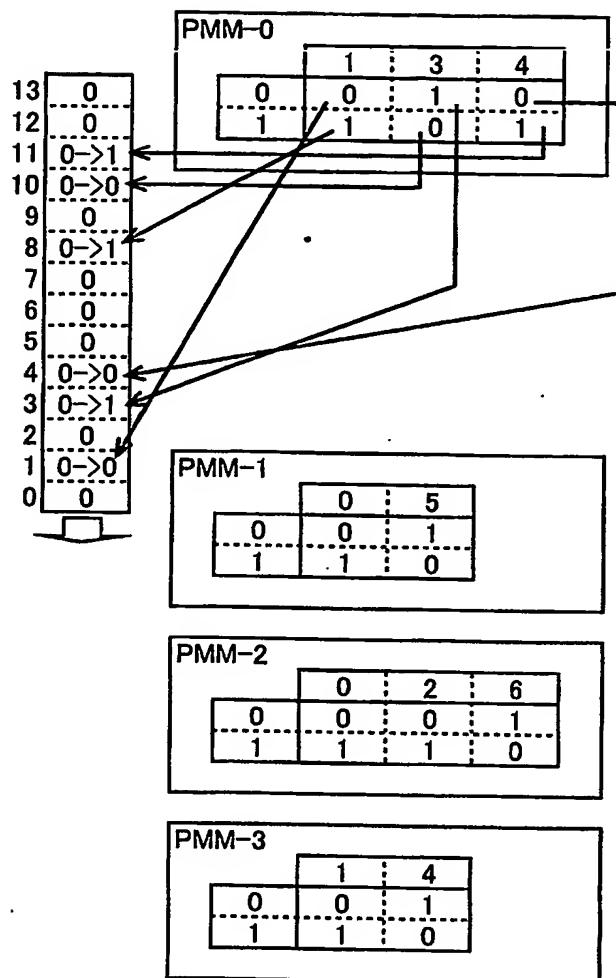


(b)



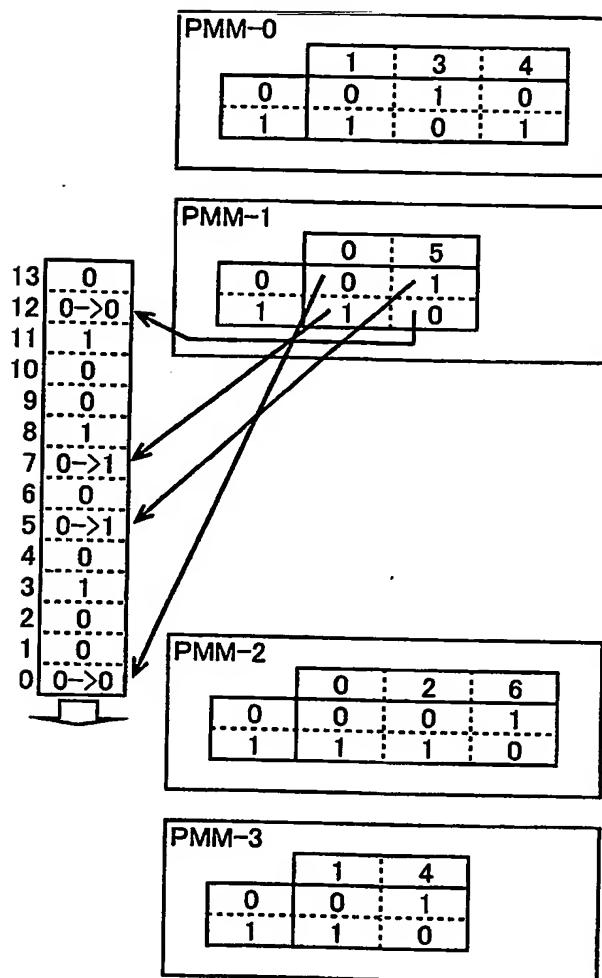
【図32】

図32



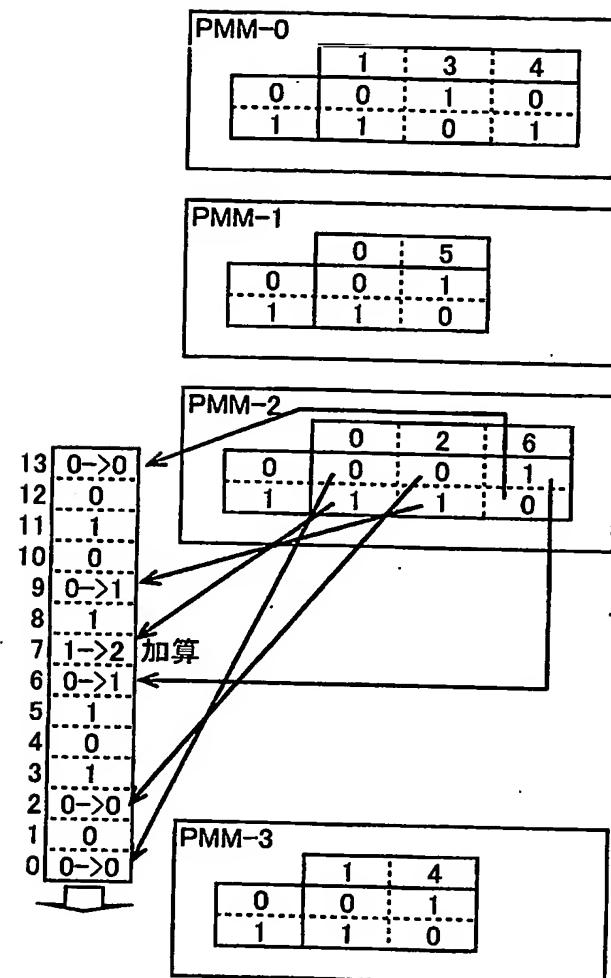
【図33】

図33



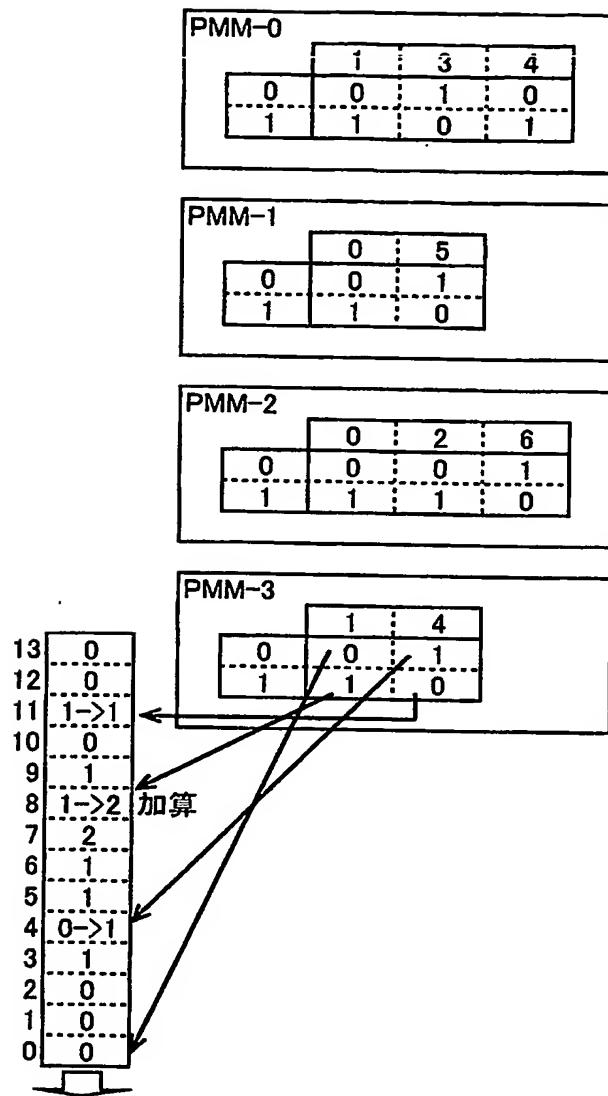
【図34】

図34



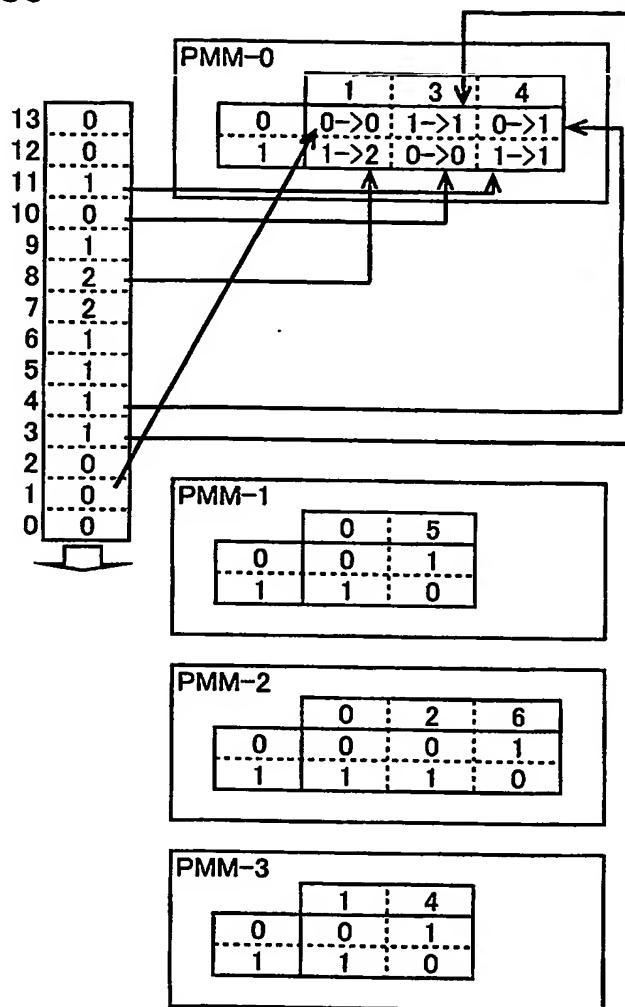
【図35】

図35



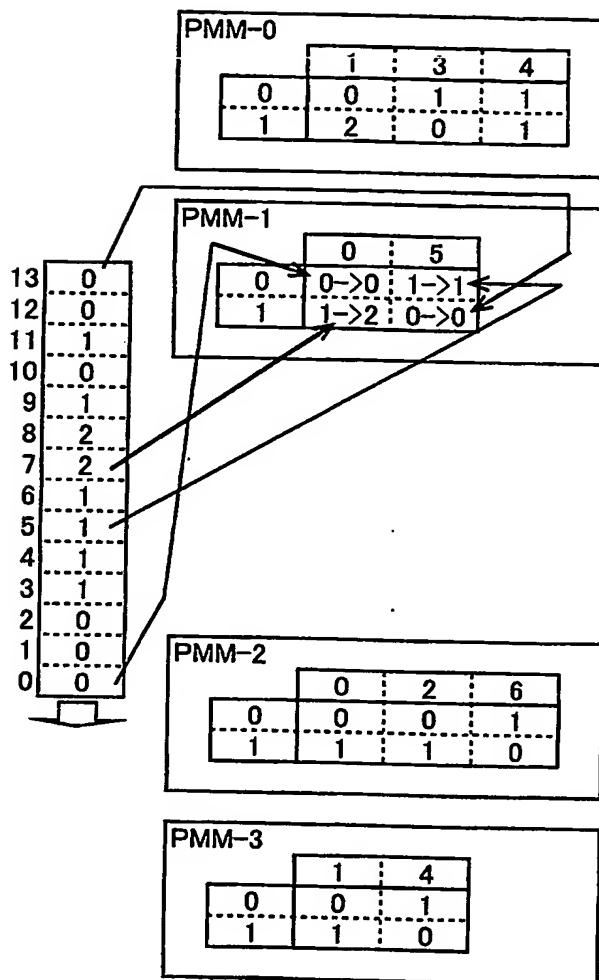
【図36】

図36



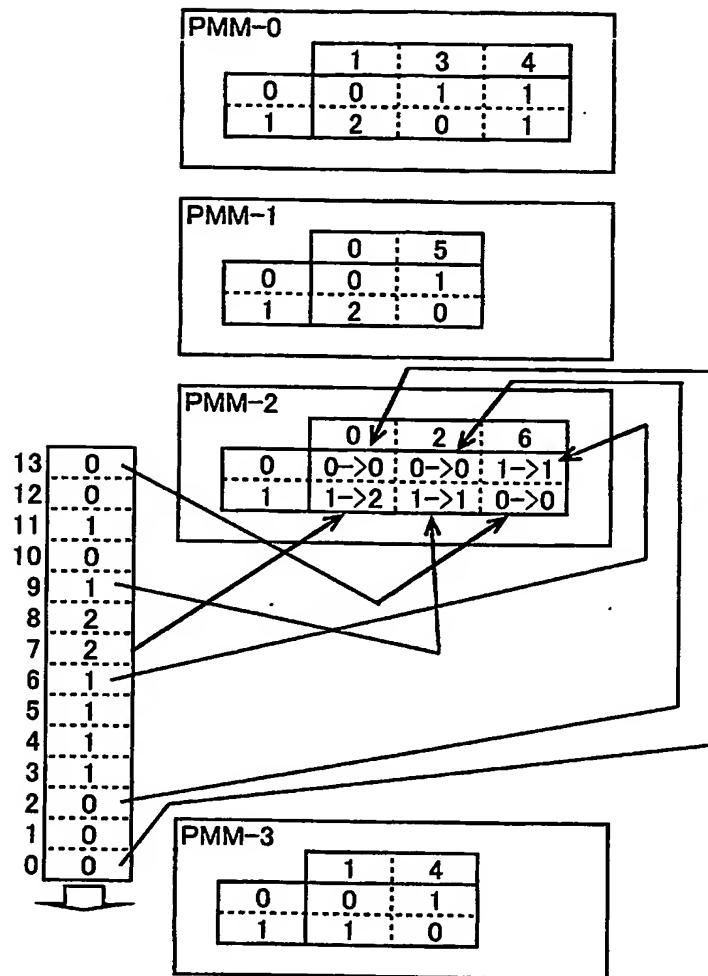
【図37】

図37



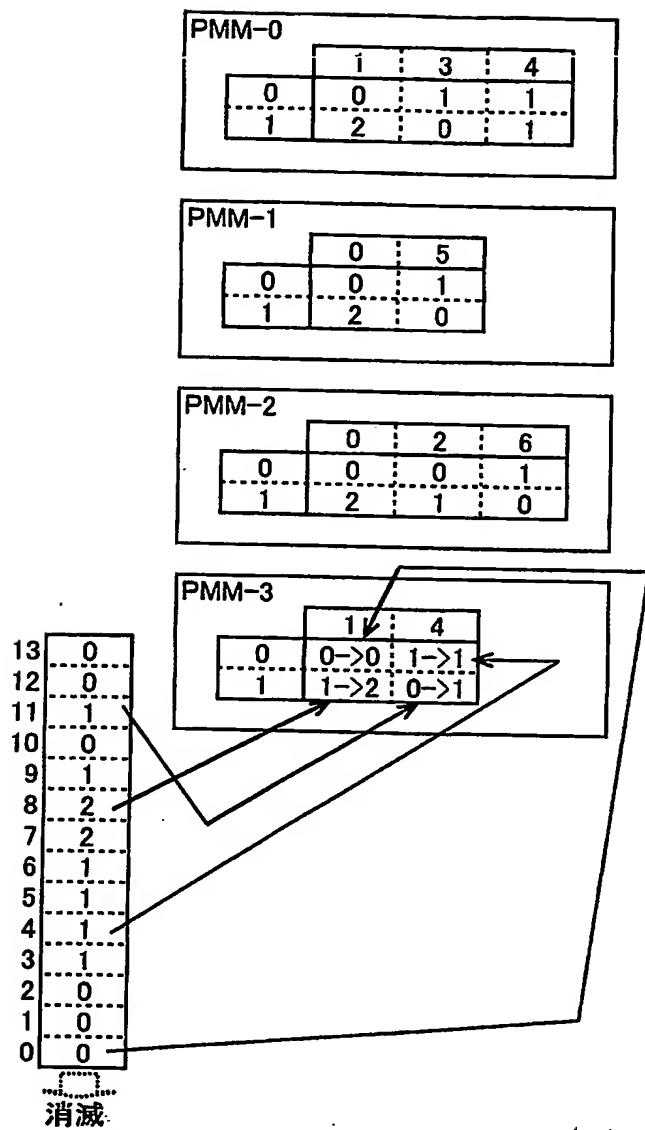
【図38】

図38



【図39】

図39



【図40】

図40

The diagram illustrates four stacked rectangular boxes, each containing a 3x3 grid table. The boxes are labeled PMM-0, PMM-1, PMM-2, and PMM-3 from top to bottom. Each box has a label number to its right: 4001 for PMM-0, 4002 for PMM-1, 4003 for PMM-2, and 4004 for PMM-3. The tables in each box have '男' (Male) in the first column and '女' (Female) in the second column. The third column is unlabeled. The numbers in the grid represent counts for three categories across three rows.

	男	女	
18	0	1	21
1	0	2	3
24	1	0	4

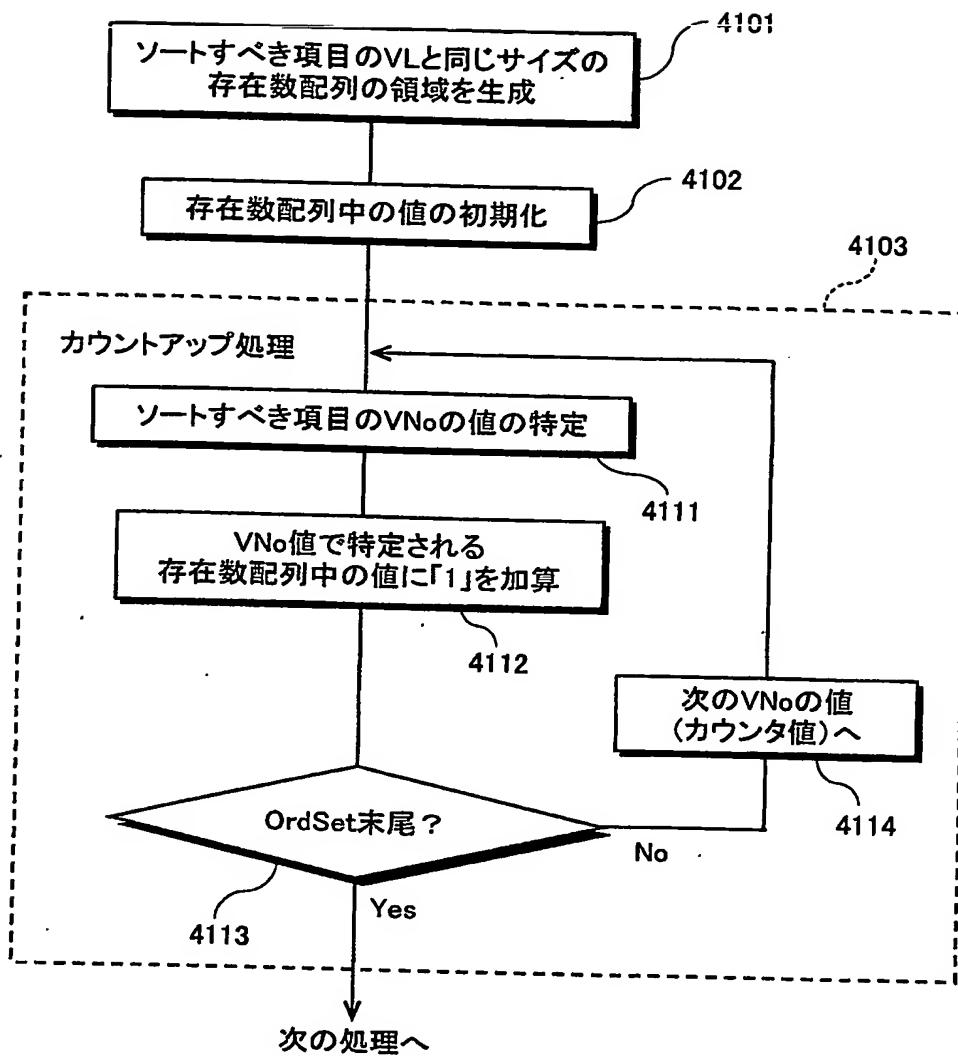
	男	女	
16	0	1	28
0	0	2	5
28	1	0	0

	男	女	
16	0	1	20
0	0	2	2
20	0	1	33

	男	女	
18	0	1	24
1	0	2	4
24	1	1	0

【図41】

図41



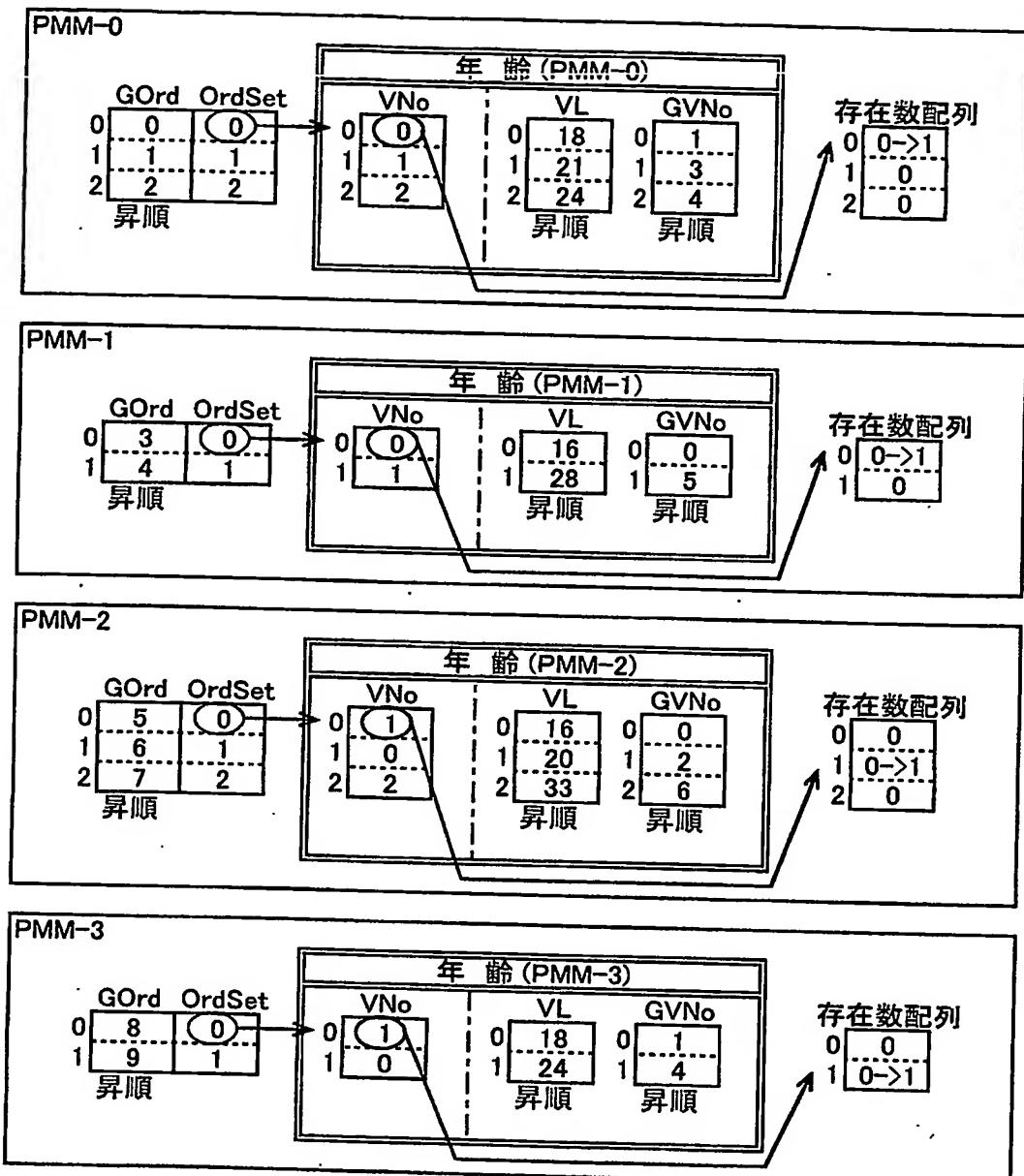
【図42】

図42

PMM-0	<table border="1"> <thead> <tr> <th>GOrd</th><th>OrdSet</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td></tr> <tr> <td>1</td><td>1</td></tr> <tr> <td>2</td><td>2</td></tr> </tbody> </table> <p>昇順</p>	GOrd	OrdSet	0	0	1	1	2	2	<table border="1"> <thead> <tr> <th colspan="3">年齢 (PMM-0)</th></tr> <tr> <th>VNo</th><th>VL</th><th>GVNo</th></tr> </thead> <tbody> <tr> <td>0 0</td><td>0 18</td><td>0 1</td></tr> <tr> <td>1 1</td><td>1 21</td><td>1 3</td></tr> <tr> <td>2 2</td><td>2 24</td><td>2 4</td></tr> </tbody> </table> <p>昇順 昇順 昇順</p>	年齢 (PMM-0)			VNo	VL	GVNo	0 0	0 18	0 1	1 1	1 21	1 3	2 2	2 24	2 4	<table border="1"> <thead> <tr> <th colspan="2">存在数配列</th></tr> </thead> <tbody> <tr> <td>0 0</td><td></td></tr> <tr> <td>1 0</td><td></td></tr> <tr> <td>2 0</td><td></td></tr> </tbody> </table>	存在数配列		0 0		1 0		2 0	
GOrd	OrdSet																																	
0	0																																	
1	1																																	
2	2																																	
年齢 (PMM-0)																																		
VNo	VL	GVNo																																
0 0	0 18	0 1																																
1 1	1 21	1 3																																
2 2	2 24	2 4																																
存在数配列																																		
0 0																																		
1 0																																		
2 0																																		
PMM-1	<table border="1"> <thead> <tr> <th>GOrd</th><th>OrdSet</th></tr> </thead> <tbody> <tr> <td>0</td><td>3</td></tr> <tr> <td>1</td><td>4</td></tr> </tbody> </table> <p>昇順</p>	GOrd	OrdSet	0	3	1	4	<table border="1"> <thead> <tr> <th colspan="3">年齢 (PMM-1)</th> </tr> <tr> <th>VNo</th> <th>VL</th> <th>GVNo</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td>0 16</td> <td>0 0</td> </tr> <tr> <td>1 1</td> <td>1 28</td> <td>1 5</td> </tr> </tbody> </table> <p>昇順 昇順</p>	年齢 (PMM-1)			VNo	VL	GVNo	0 0	0 16	0 0	1 1	1 28	1 5	<table border="1"> <thead> <tr> <th colspan="2">存在数配列</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td></td> </tr> <tr> <td>1 0</td> <td></td> </tr> </tbody> </table>	存在数配列		0 0		1 0								
GOrd	OrdSet																																	
0	3																																	
1	4																																	
年齢 (PMM-1)																																		
VNo	VL	GVNo																																
0 0	0 16	0 0																																
1 1	1 28	1 5																																
存在数配列																																		
0 0																																		
1 0																																		
PMM-2	<table border="1"> <thead> <tr> <th>GOrd</th><th>OrdSet</th></tr> </thead> <tbody> <tr> <td>0</td><td>5</td></tr> <tr> <td>1</td><td>6</td></tr> <tr> <td>2</td><td>7</td></tr> </tbody> </table> <p>昇順</p>	GOrd	OrdSet	0	5	1	6	2	7	<table border="1"> <thead> <tr> <th colspan="3">年齢 (PMM-2)</th> </tr> <tr> <th>VNo</th> <th>VL</th> <th>GVNo</th> </tr> </thead> <tbody> <tr> <td>0 1</td> <td>0 16</td> <td>0 0</td> </tr> <tr> <td>1 0</td> <td>1 20</td> <td>1 2</td> </tr> <tr> <td>2 2</td> <td>2 33</td> <td>2 6</td> </tr> </tbody> </table> <p>昇順 昇順</p>	年齢 (PMM-2)			VNo	VL	GVNo	0 1	0 16	0 0	1 0	1 20	1 2	2 2	2 33	2 6	<table border="1"> <thead> <tr> <th colspan="2">存在数配列</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td></td> </tr> <tr> <td>1 0</td> <td></td> </tr> <tr> <td>2 0</td> <td></td> </tr> </tbody> </table>	存在数配列		0 0		1 0		2 0	
GOrd	OrdSet																																	
0	5																																	
1	6																																	
2	7																																	
年齢 (PMM-2)																																		
VNo	VL	GVNo																																
0 1	0 16	0 0																																
1 0	1 20	1 2																																
2 2	2 33	2 6																																
存在数配列																																		
0 0																																		
1 0																																		
2 0																																		
PMM-3	<table border="1"> <thead> <tr> <th>GOrd</th><th>OrdSet</th></tr> </thead> <tbody> <tr> <td>0</td><td>8</td></tr> <tr> <td>1</td><td>9</td></tr> </tbody> </table> <p>昇順</p>	GOrd	OrdSet	0	8	1	9	<table border="1"> <thead> <tr> <th colspan="3">年齢 (PMM-3)</th> </tr> <tr> <th>VNo</th> <th>VL</th> <th>GVNo</th> </tr> </thead> <tbody> <tr> <td>0 1</td> <td>0 18</td> <td>0 1</td> </tr> <tr> <td>1 0</td> <td>1 24</td> <td>1 4</td> </tr> </tbody> </table> <p>昇順 昇順</p>	年齢 (PMM-3)			VNo	VL	GVNo	0 1	0 18	0 1	1 0	1 24	1 4	<table border="1"> <thead> <tr> <th colspan="2">存在数配列</th> </tr> </thead> <tbody> <tr> <td>0 0</td> <td></td> </tr> <tr> <td>1 0</td> <td></td> </tr> </tbody> </table>	存在数配列		0 0		1 0								
GOrd	OrdSet																																	
0	8																																	
1	9																																	
年齢 (PMM-3)																																		
VNo	VL	GVNo																																
0 1	0 18	0 1																																
1 0	1 24	1 4																																
存在数配列																																		
0 0																																		
1 0																																		

【図43】

図43



【図44】

図44

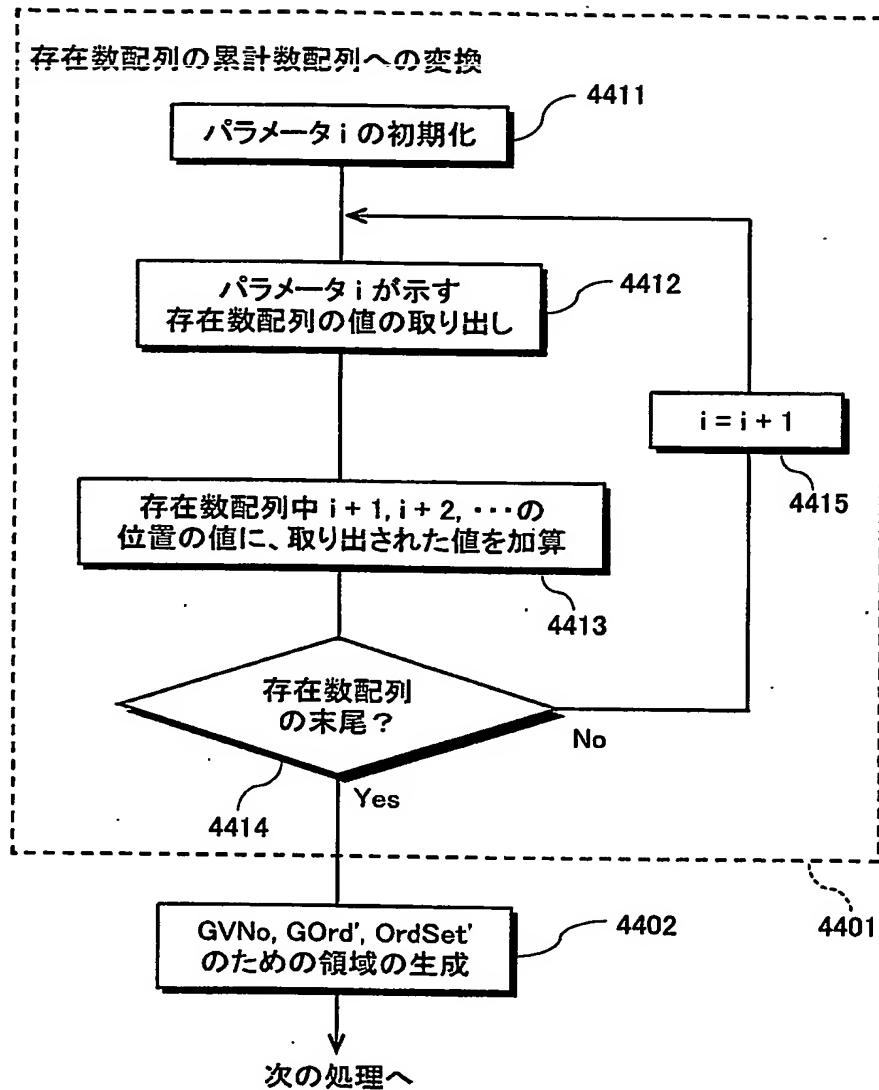
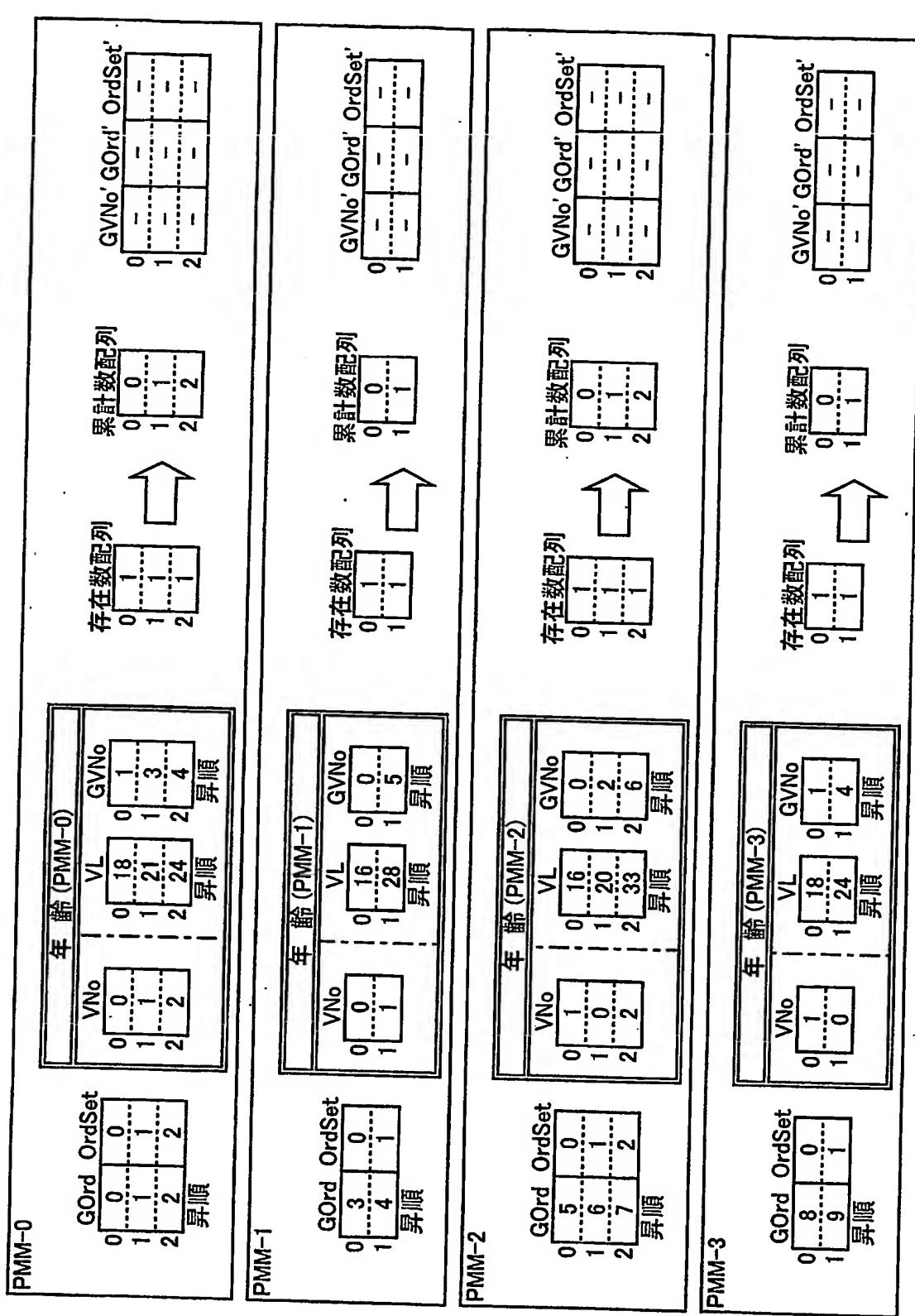


図45



【図46】

図46

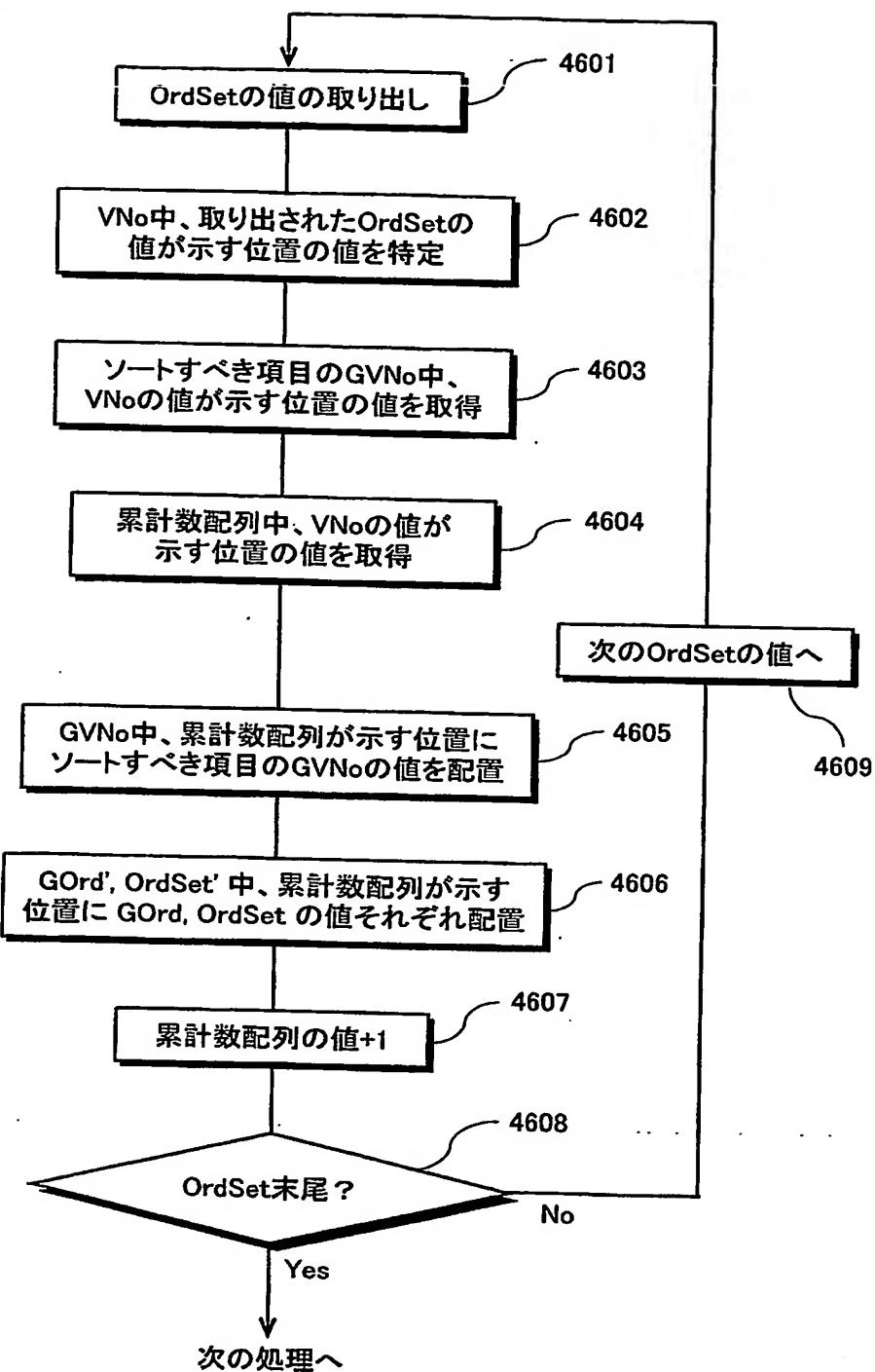


図47

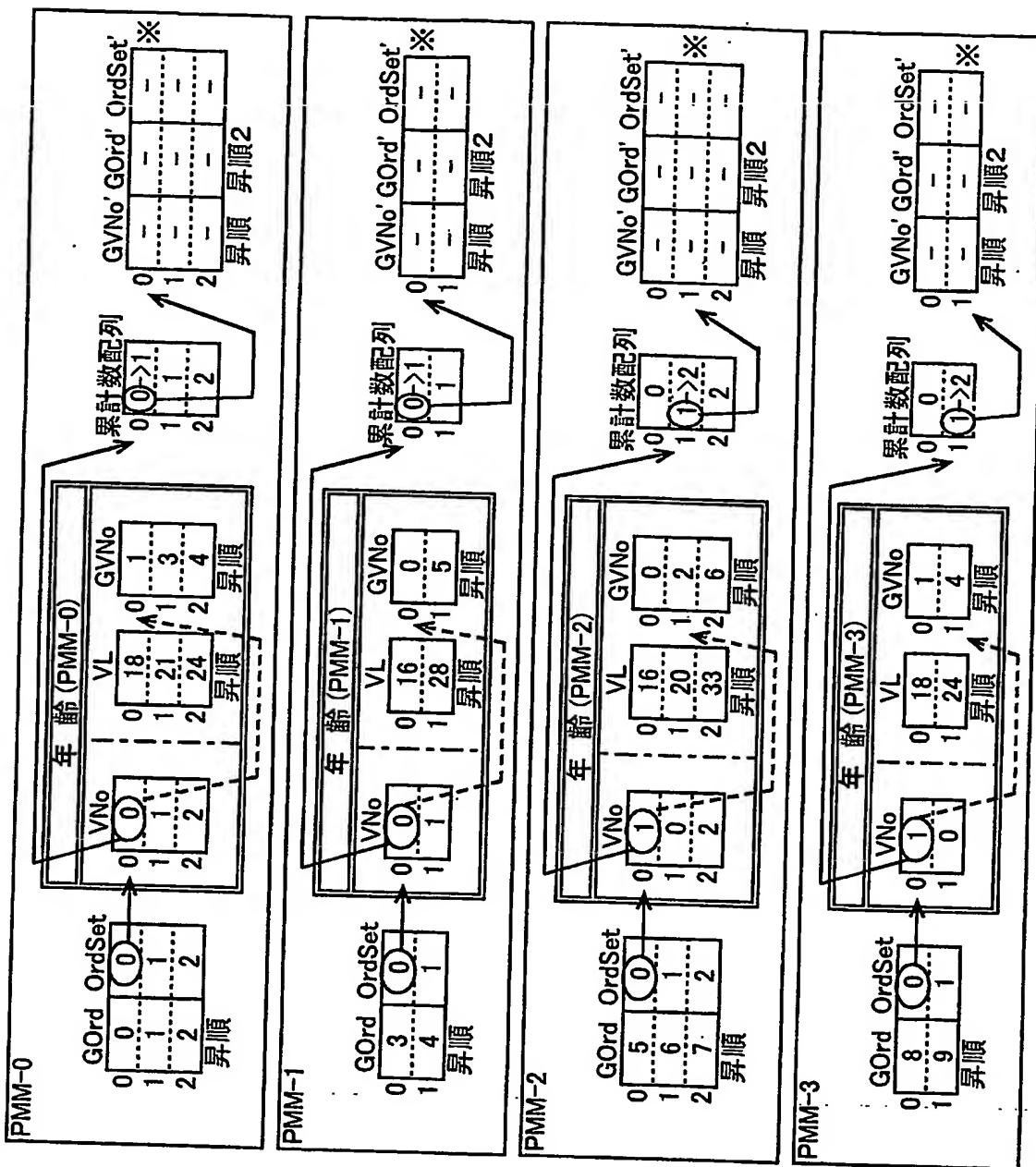


図48】

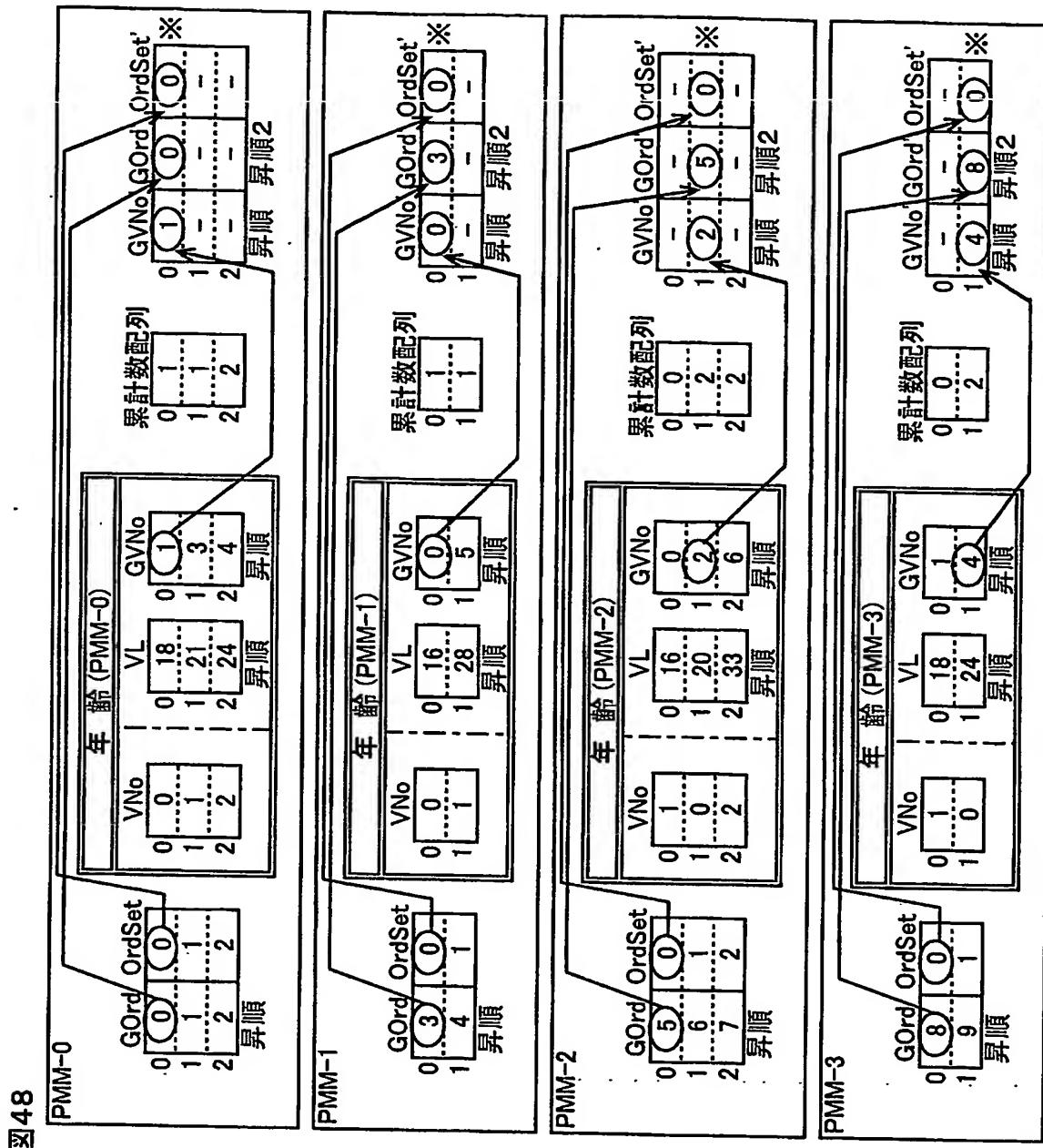
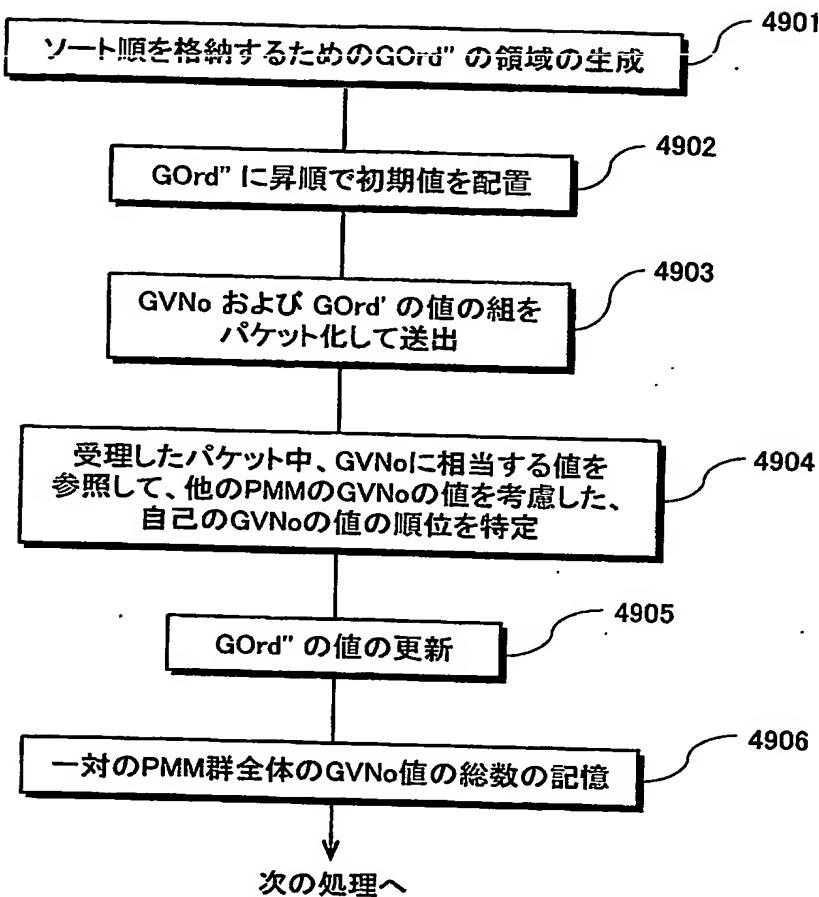


図48

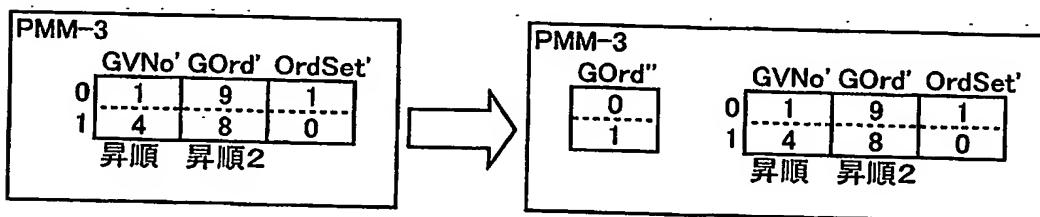
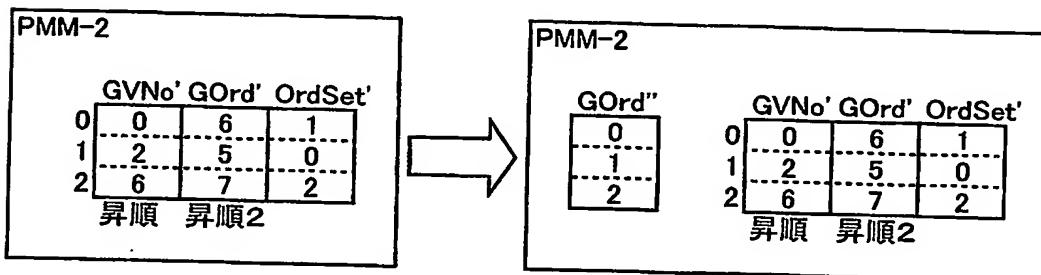
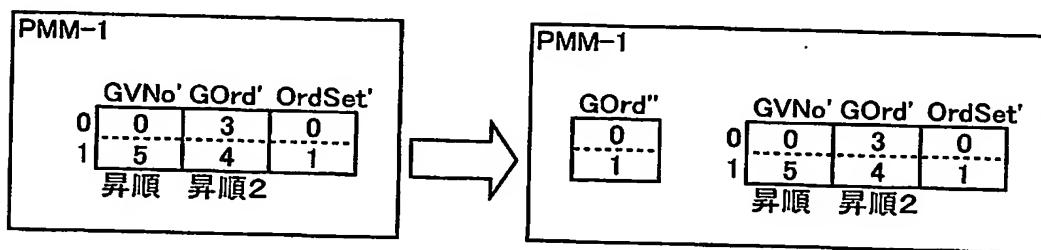
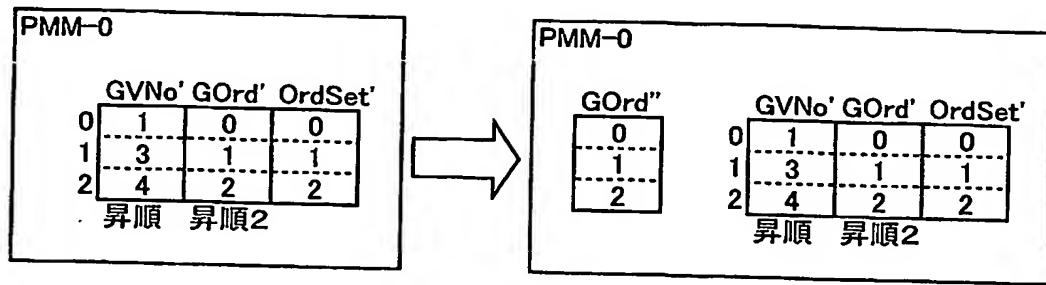
【図49】

図49



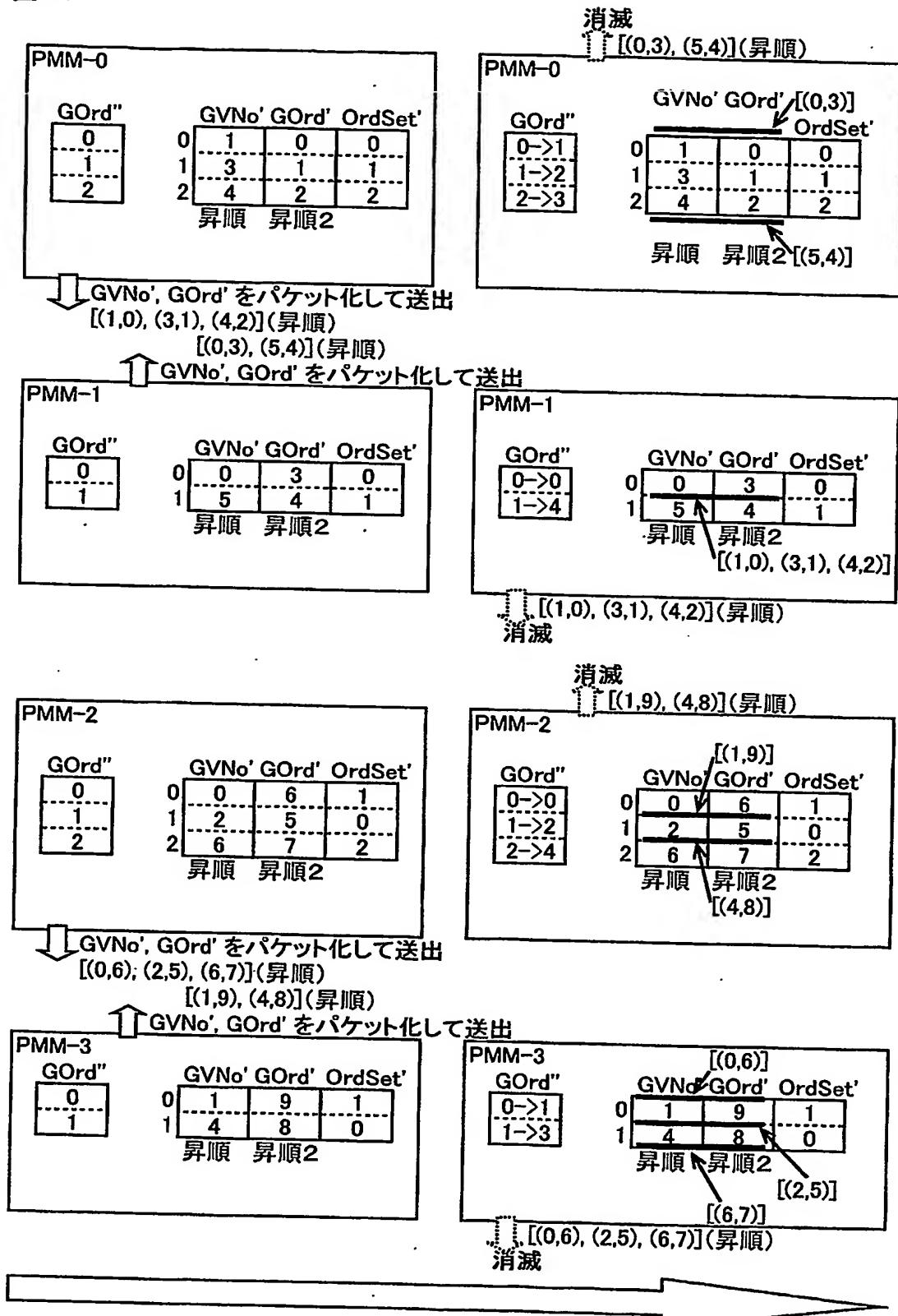
【図50】

図50



【図 51】

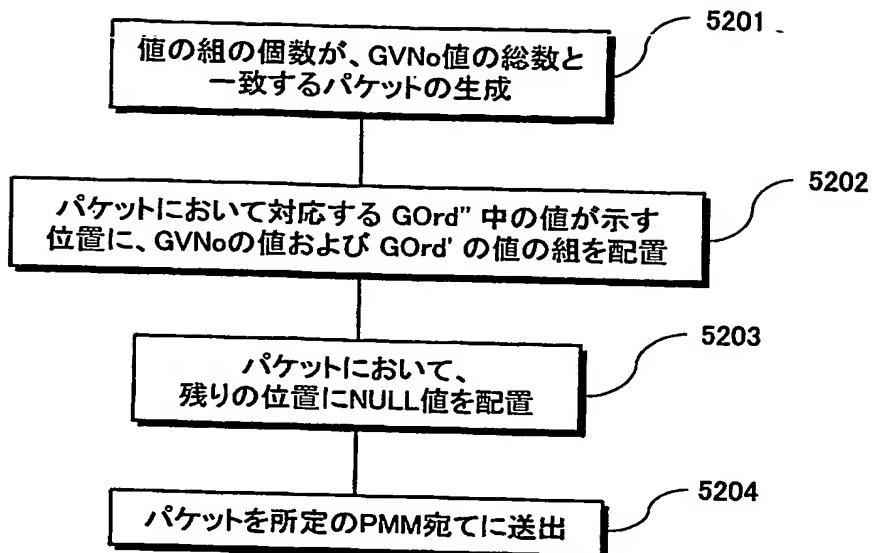
図51



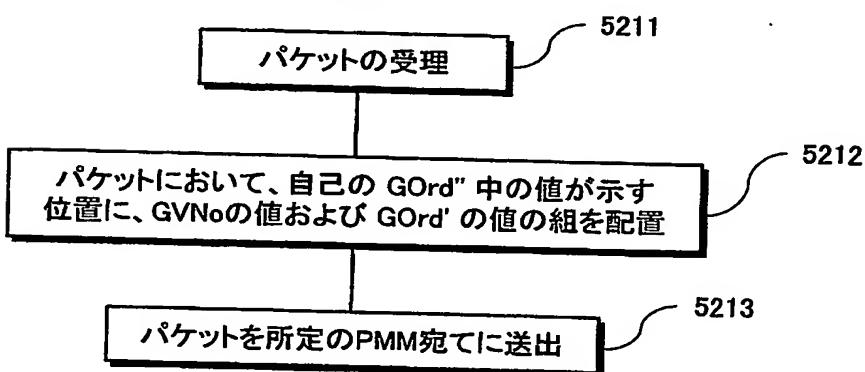
【図52】

図52

(a)

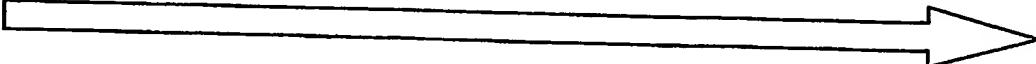
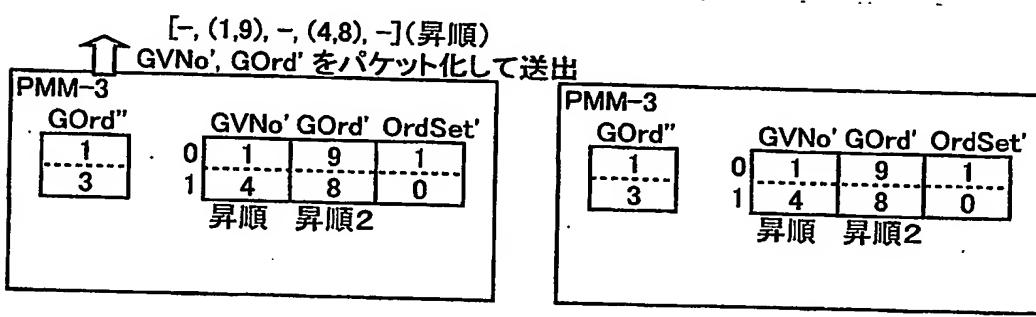
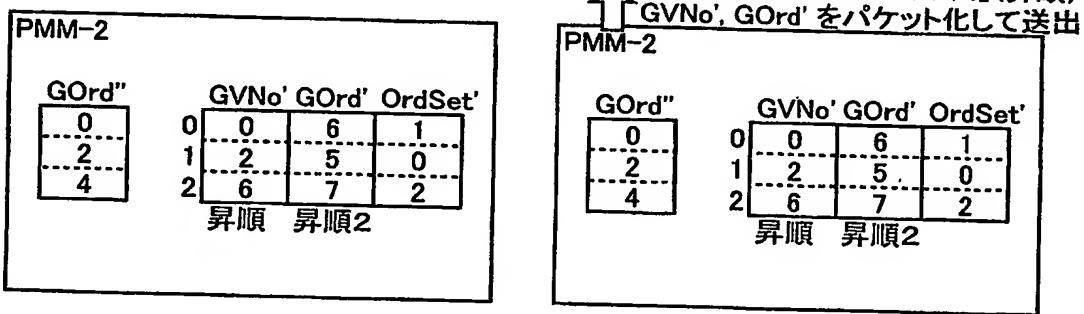
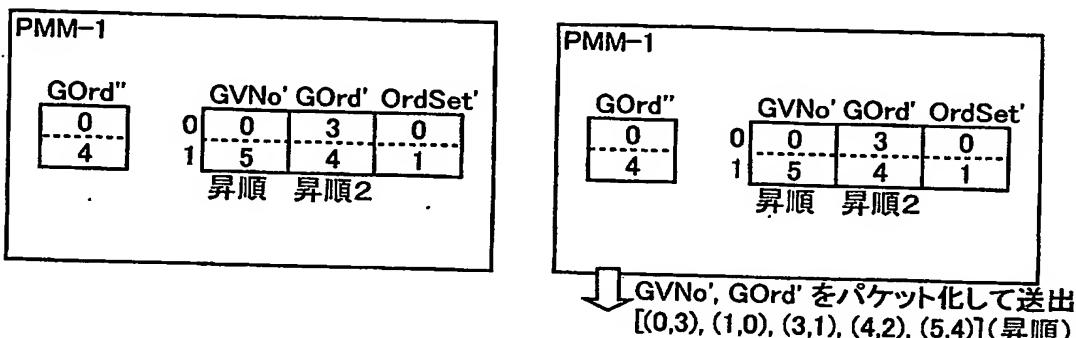
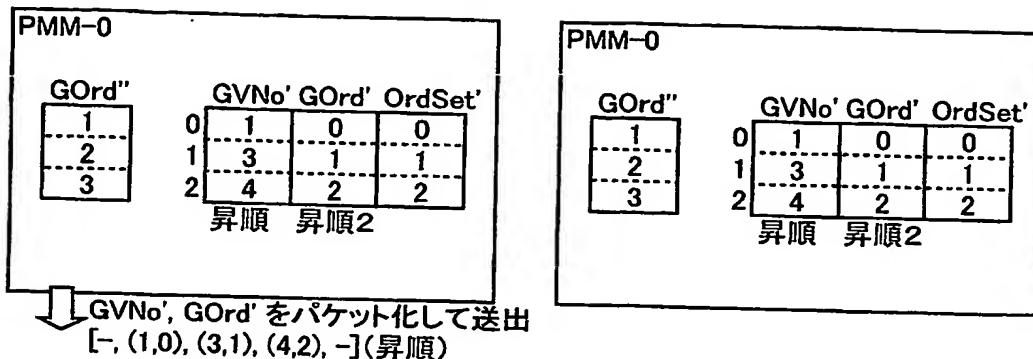


(b)



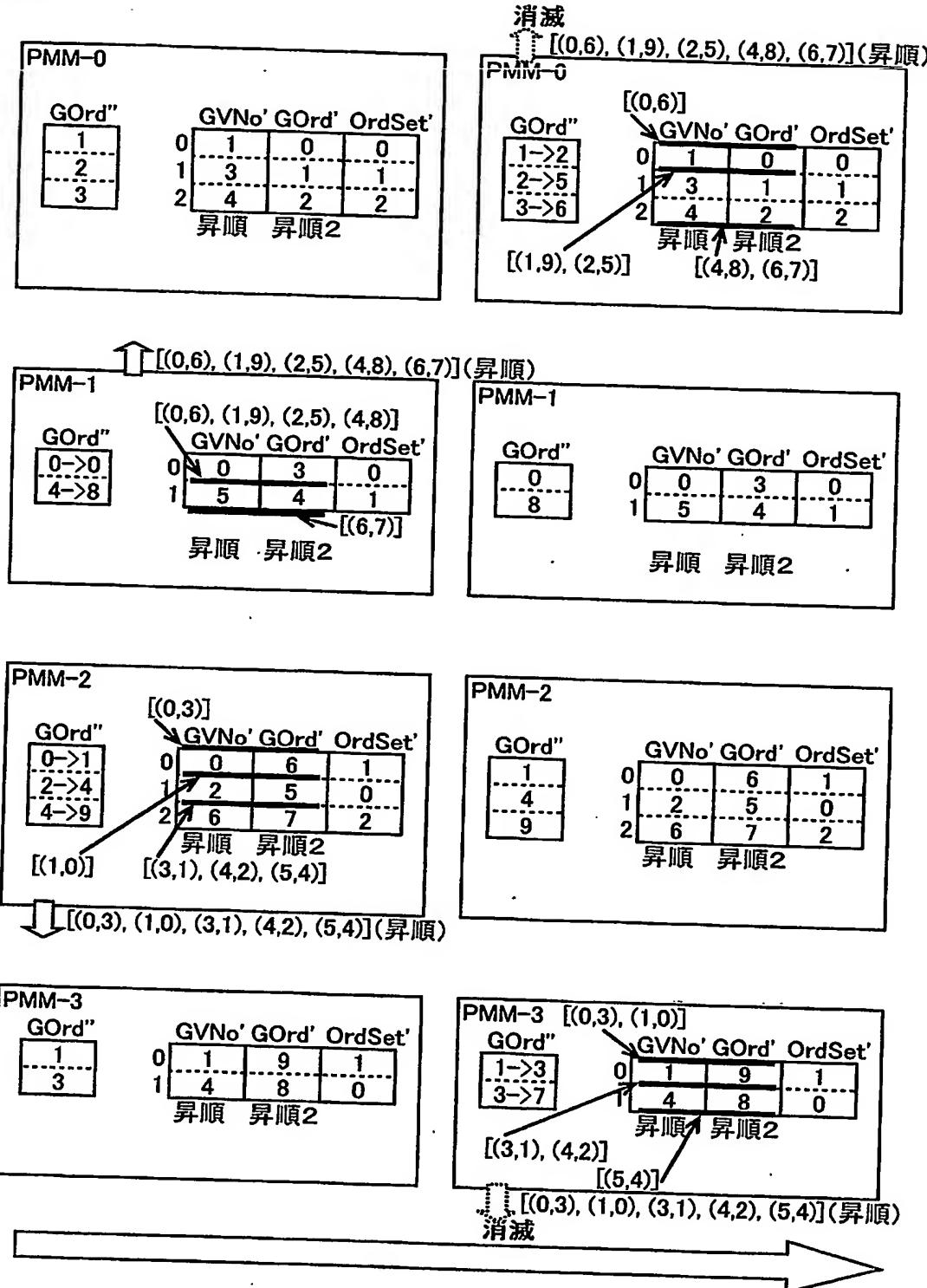
【図 53】

図53



【図 54】

図54



【図55】

図55

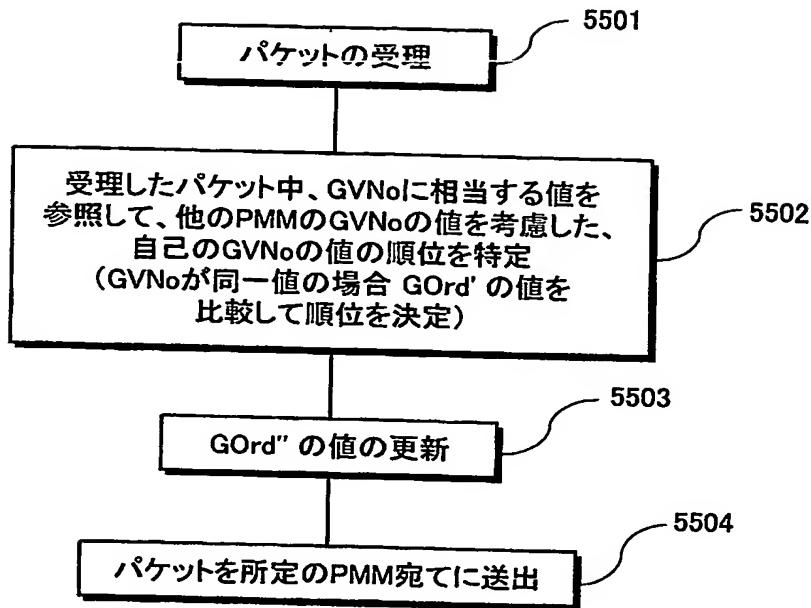


図56

性別(PMM-0)				年齢(PMM-0)				身長(PMM-0)				体重(PMM-0)				
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo
0 2	0	0 1	0 男	0 0	0 0	1 8	0 1	0 159	0 0	0 48	0 0	1 55	1 2	2 64	2 3	昇順
1 5	1	1 0	1 女	1 1	1 1	2 1	1 3	1 168	1 2	1 55	1 2	2 0	2 1	2 64	2 3	昇順
2 6	2	2 1			2 2	2 4	2 4	2 172	2 4	2 0	2 0	2 0	2 0	2 64	2 3	昇順
OFFSET=0																
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo
0 0	0	0 1	0 男	0 0	0 0	1 6	0 5	0 172	0 4	0 48	0 0	1 78	1 5	1 78	1 5	昇順
1 8	1	1 0	1 女	1 1	1 1	2 8	1 5	1 181	1 7	1 78	1 5	2 0	2 0	2 0	2 0	昇順
OFFSET=3																
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo
0 0	0	0 1	0 男	0 0	0 0	1 6	0 5	0 172	0 4	0 48	0 0	1 78	1 5	1 78	1 5	昇順
1 8	1	1 0	1 女	1 1	1 1	2 8	1 5	1 181	1 7	1 78	1 5	2 0	2 0	2 0	2 0	昇順
OFFSET=5																
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo
0 1	1	0 1	0 男	0 0	0 1	1 6	0 5	0 166	0 1	0 52	0 1	1 55	1 2	2 65	2 4	昇順
1 4	0	1 1	1 女	1 1	1 0	2 0	1 2	1 168	1 2	1 0	1 0	1 0	1 0	1 0	1 0	昇順
2 9	2	2 0			2 2	2 33	2 6	2 174	2 5	2 2	2 2	2 2	2 2	2 2	2 2	昇順
OFFSET=8																
GOrd	OrdSet	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo	VNo	VL	GVNo
0 3	1	0 0	0 男	0 0	0 1	1 8	0 4	0 170	0 3	0 55	0 2	1 64	1 3	1 64	1 3	昇順
1 7	0	1 1	1 女	1 1	1 0	1 24	1 4	1 177	1 6	1 0	1 0	1 0	1 0	1 0	1 0	昇順

【図57】

図57

	性別	年齢	身長	体重
0	女	16	168	55
1	男	21	172	64
2	女	24	159	48
3	女	16	172	48
4	男	28	181	78
5	女	20	166	55
6	女	16	168	52
7	男	33	174	65
8	男	24	177	64
9	女	18	170	55

ソート前

	性別	年齢	身長	体重
0	3 女	16	172	48
1	6 女	16	168	52
2	0 女	18	168	55
3	9 女	18	170	55
4	5 女	20	166	55
5	1 男	21	172	64
6	2 女	24	159	48
7	8 男	24	177	64
8	4 男	28	181	78
9	7 男	33	174	65

ソート後

【図58】

5800

	性別	年齢	身長(cm)	体重(kg)
0	女	18	168	55
1	男	21	172	64
2	女	24	159	48
3	女	16	172	48
4	男	28	181	78
5	女	20	166	55
6	女	16	168	52
7	男	33	174	65
8	男	24	177	64
9	女	18	170	55

	OrdSet
0	3
1	6
2	0
3	9
4	5
5	1
6	2
7	8
8	4
9	7

↑

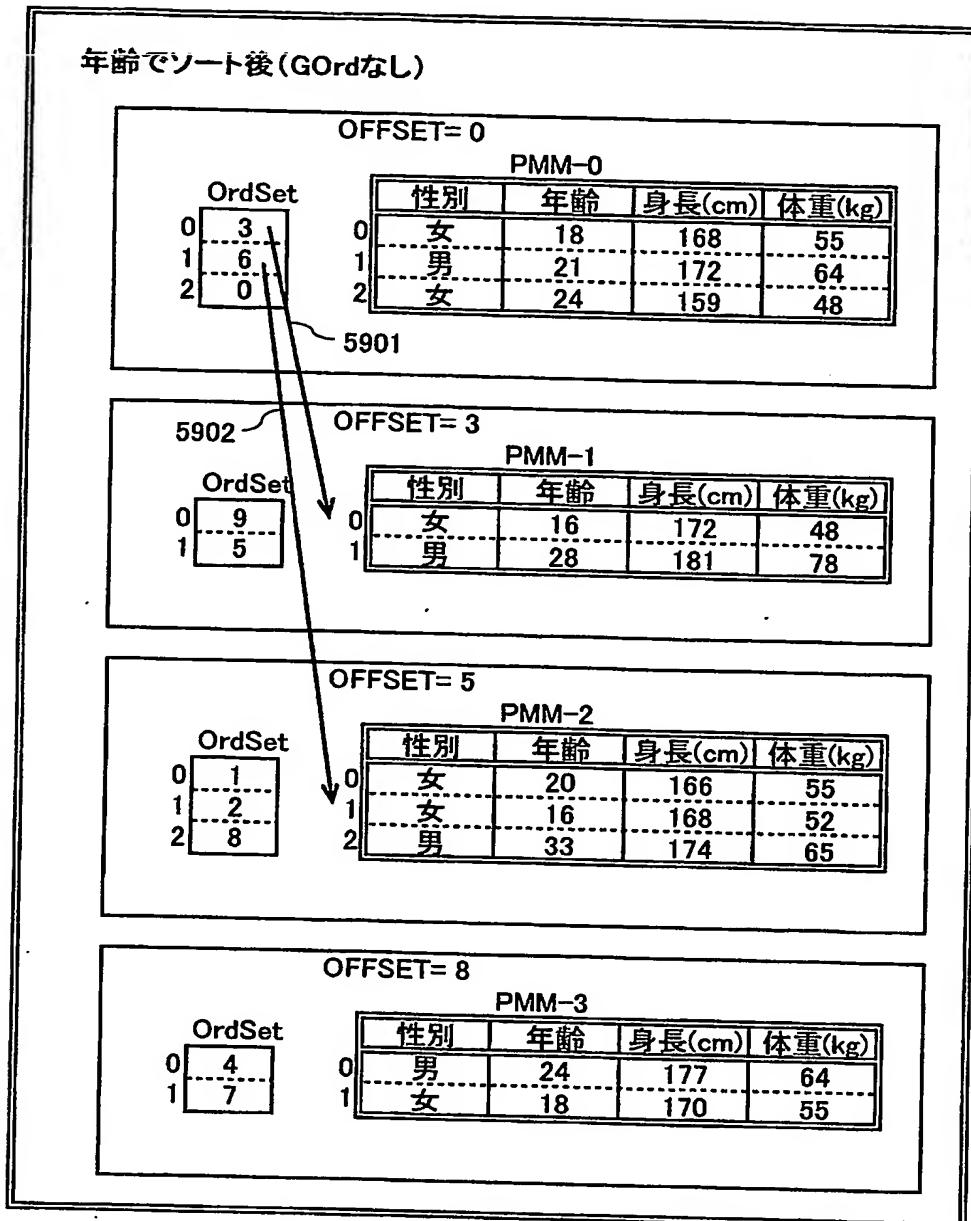
	性別	年齢	身長(cm)	体重(kg)
0	女	18	168	55
1	男	21	172	64
2	女	24	159	48
3	女	16	172	48
4	男	28	181	78
5	女	20	166	55
6	女	16	168	52
7	男	33	174	65
8	男	24	177	64
9	女	18	170	55

	OrdSet
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9

図58

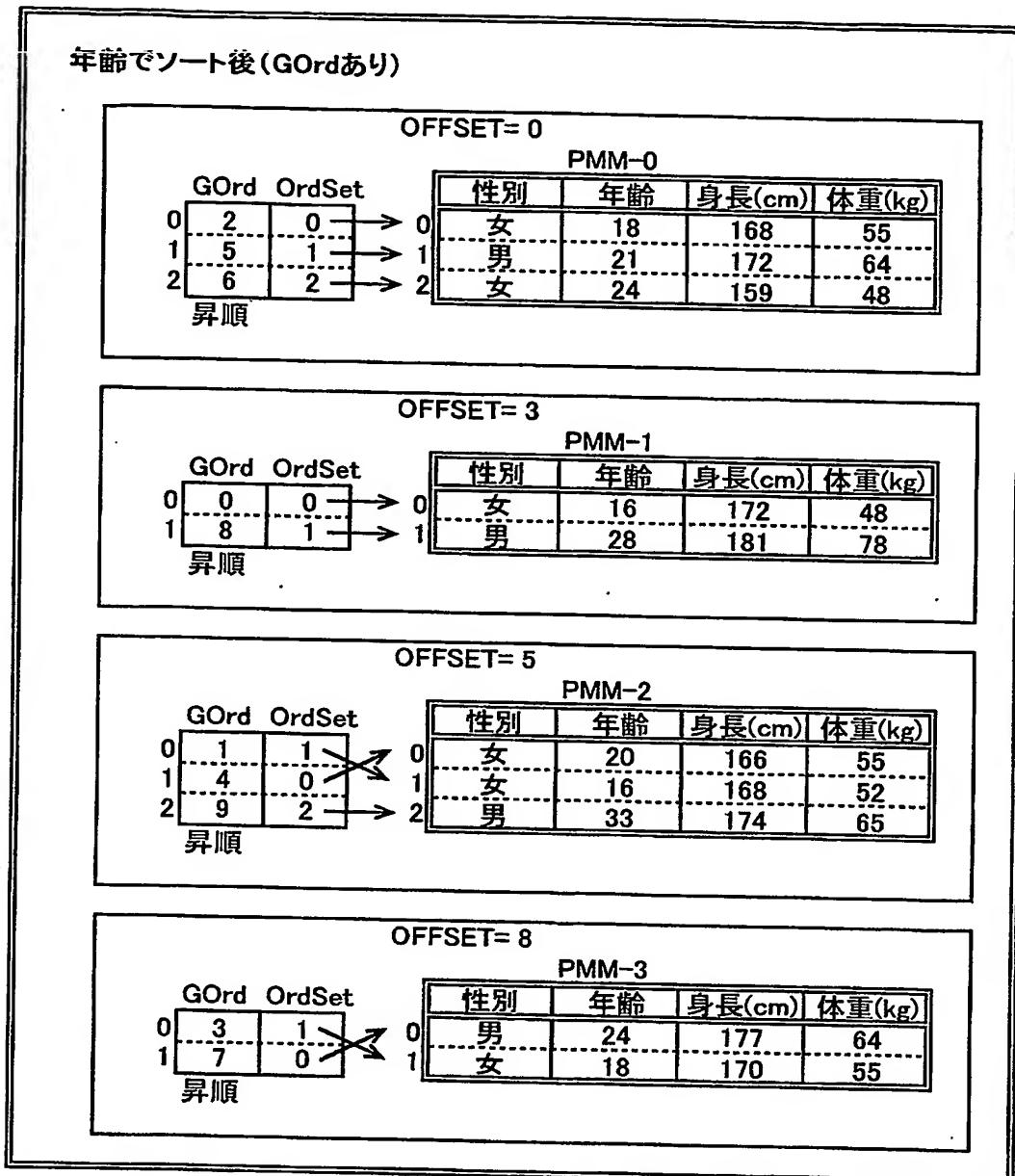
【図59】

図59



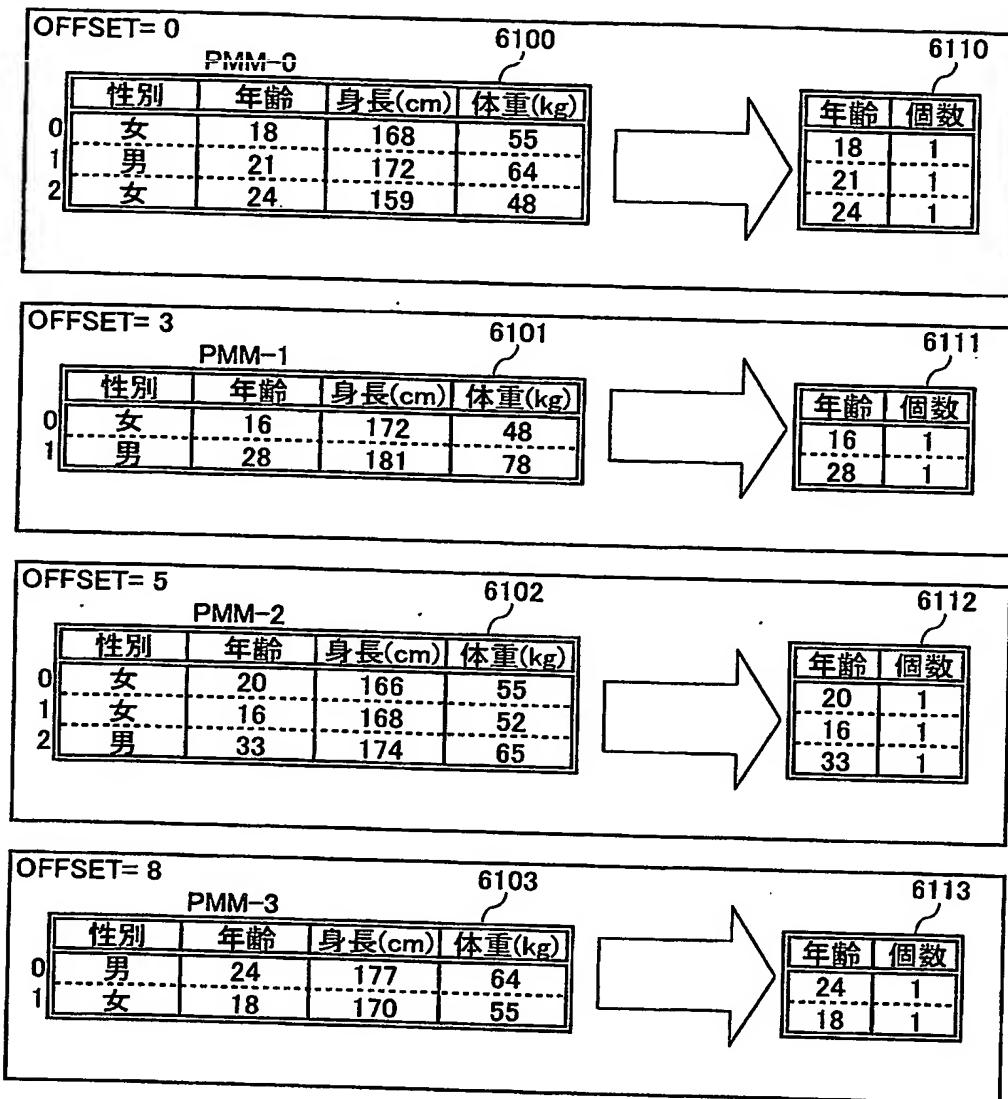
【図60】

図60



【図61】

図61



【書類名】 要約書

【要約】

【課題】 分散メモリ型において、单一命令により種々のメモリに記憶された配列中の要素を入出力し、処理と通信を統合する。

【解決手段】 隣接するPMM12の間には、一方から他方にパケットを伝達する第1のパケット伝送路14、および、他方から一方にパケットを伝達する第2のパケット伝送路16を備える。各PMMは、各々が項目と当該項目に属する項目値とを含むレコードの配列として表される表形式データを表現するための、特定の項目に属する項目値に対応した項目値番号の順に当該項目値が格納されている値リスト、および、一意的な順序集合配列の順に、当該項目値番号を指示するためのポインタ値が格納されたポインタ配列からなる情報ブロックを保持し、各メモリにて保持された情報ブロックの集合体により、グローバルな情報ブロックが形成される。

【選択図】 図1

認定・付加情報

特許出願の番号 特願2003-111978
受付番号 50300632264
書類名 特許願
担当官 第七担当上席 0096
作成日 平成15年 4月21日

<認定情報・付加情報>

【提出日】 平成15年 4月16日

次頁無

出証特2004-3045031

特願 2003-111978

出願人履歴情報

識別番号 [598108515]

1. 変更年月日 1998年 8月11日

[変更理由] 新規登録

住 所 神奈川県横浜市神奈川区松見町4丁目1101番地7 コート

ハウス菊名804号

氏 名 古庄 晋二